

CLUB I.S.I.S.



AÑO 1º - Nº 2 y 3 NOVIEMBRE-DICIEMBRE 1984

PROMUEVE Y EDITA
I.S.I.S.S.A.
c/ Lagasca, 125
Tlfno. 411 55 61
411 55 43

IMPRIME: ARTYDIS
Gabriel Lobo, 10
Telfno. 411 44 10
28002 MADRID

D.L. M. 165-1985

COLABORARON

JAHAIRA PEREZ
INAKI CASTILLO
Fco. J. DE LOS DOLORES
FRANCISCO J. GARCIA
IGNACIO JIMENEZ

INDICE

/DAT
prime
idad. En
pletas, es
líneas DATA
as líneas. C
es.
ión escribimos
100 FOR I= 6
10 READ A:
al hacer
es así es q
a última prueb
vez hecho todo e
r ahí y la po
siguiente coman
BSAVE "CARAC
(Para cargar
cemos con esto es
eres redefinidos
s.
uación viene la pru
nstrucciones: (debe cu
:")
VP 815, V
A Ø.=
primer
ado y
abíam

FELIZ AÑO 1985

De nuevo estamos con vosotros y hemos cubierto una nueva etapa en la elaboración de este boletín.

Ello nos ha supuesto un mayor esfuerzo y en consecuencia algún retraso, pero creemos que merece la pena pues todo va enfocado a lograr una mejor calidad tanto tecnicamente como en presentación.

En fin, seguimos en la brecha y tratando de conseguir un nivel informativo de calidad, aunque os recordamos que para ésto necesitamos más colaboración por parte de los socios, en cuanto a programas y sugerencias.



I S I S ha puesto a la venta una BASE DE DATOS para CASSETTE y 328, que es el comienzo de una serie de programas y juegos, enfocados a SPECTRAVIDEO, totalmente confeccionados por nosotros.

Una vez examinado el mercado de software a este nivel, hemos llegado a la conclusión de que en muchas casos la calidad no es la más apropiada, por lo tanto vamos a lanzar una serie de programas que deseamos se distingan por su calidad.

Esperamos la colaboración de todos los socios que quieran comercializar programas de interés general.

BASE DE DATOS ISIS (cassette)

Este programa permite que el usuario se diseñe un fichero en memoria compuesto de registros cuyo formato ha diseñado previamente.

Dicho fichero es almacenado en su totalidad en la memoria,por lo que el número de registros depende de la cantidad de campos de los mismos.

Una vez creado el fichero, con sus datos correspondientes, se puede modificar, listar u organizar por cualquier campo, grabándolo en la cassette, cuando se desee.

Si disponemos de impresora podremos realizar listados en papel.

Lógicamente, por razones de capacidad no podemos tener en memoria nada más que un fichero simultáneamente, pero no obstante, este programa puede solucionar bastantes casos de pequeños ficheros, que deseemos organizar por diferentes criterios.

El precio de este programa es de 4.500,- Ptas. para la calle y de 3.500,-Ptas. para los socios del CLUB

Para los socios del Club que traduzcan esto :

45	4C	20	43	4C	55	42	20	48	41	52	41
20	55	4E	20	31	30	25	44	45	20	44	45
53	43	55	45	4E	54	4F	20	45	4E	20	54
4F	44	4F	53	20	53	55	53	20	41	52	54
49	43	55	4C	4F	53	2E	0D	0A			

SOFTWARE SPECTRAVIDEO	P.V.P./Ptas.
- SD-220C SECTOR ALPHA	4.900,-
- SD-232C FRANTIC FREDDY	7.500,-
- SD-236C MUSIC MENTOR	4.900,-
- SD-237C SUPER CROSS FORCE	3.995,-
- SD-291C FLIPER SLIPPER	3.995,-
- SD-294C JUSTWRITE JR.	6.800,-
- SD-101T M.GOLF	2.300,-
- SD-211T OLD MAC FARMER	2.300,-
- SD-216T TETRA HORROR	2.300,-
- SD-219T NEW YORK BOMB BLITZ	2.300,-
- SD-227T SPECTRA CHECKBOOK	2.300,-
- SD-231T ARTIST	2.300,-
- SD-233T ARMOURED ASSAULT	2.300,-
- SD-234T SPECTRON	2.300,-
- SD-235T INTRODUCCION AL BASIC	2.300,-
- SD-238T OTELLO	2.300,-
- SD-241T TELE BUNNY	2.300,-
- SD-242T TURBOAT	2.300,-
- SD-243T SASA	2.300,-
- SD-266T SPRITE EDITOR	2.300,-
- SD-292T NINJA	2.300,-
- SD-221T SWING MAN	2.300,-
- SD-293T KUNG FU MASTER	2.300,-
- SD-301T BIORITMO	2.300,-
- SD-305T BOA	2.300,-
- SD- T RESCATE	2.300,-
- SD- T SQUASH	2.300,-
- COBOL-80 (D)	170.000,-
- DBASE II (D)	112.000,-
- FORTH-79 (D)	23.950,-
- FORTRAN (D)	84.200,-
- SUPERCALC (D)	25.795,-
- SUPERCALC 2 (D)	58.330,-
- TINY PASCAL (D)	19.045,-
- TURBO PASCAL (D)	17.855,-
- WORDSTAR (D)	98.000,-
- BASE DE DATOS CANTELLI (T)	4.900,-

SOFTWARE KAYPRO Y SPECTRAVIDEO (PARA DISCO)	P.V.P./Ptas.
- CONTABILIDAD ver. 1.1.	60.000,-
- STOCK	30.000,-
- ABOGADOS	60.000,-
- MEDICOS	50.000,-
- HORAS	40.000,-
- EL PROGRAMA DE MEDICOS Y HORAS JUNTOS	80.000,-
- GESTION DE INMOBILIARIAS	50.000,-
- ADMINISTRACION DE FINCAS	50.000,-
- BIBLIOTECAS (en preparacion)	
- OPTICOS "	
- ODONTOLOGOS "	
- OFTALMOLOGOS "	
- OTORRINOS "	

MONITORES	
- MONITOR DYNADATA 12" FOSFORO VERDE CON SONIDO	21.900,-
- MONITOR DYNADATA 12" FOSFORO VERDE CON SONIDO Y PANTALLA ORIENTABLE	29.900,-

ACCESORIOS	
- DISCO DE UNA CARA SIMPLE DENSIDAD (MF 11)	430,-
- DISCO DE UNA CARA DOBLE DENSIDAD (MF21)	489,-
- DISCO DE DOBLE CARA DOBLE DENSIDAD (MF22)	620,-

LISTA DE PRECIOS VENTA AL PUBLICO

UNIDADES CENTRALES	P.V.P./Ptas.
- KAYPRO II (400 Kb.)........................	405.000,-
- KAYPRO IV (800 Kb.)........................	525.000,-
- KAYPRO X (10.5 Mb.)......................	810.000,-
- FUTURE FX15 (800 Kb).......................	460.000,-
- FUTURE FX20 (1600 Kb.).....................	495.000.-
- FUTURE FX30/5 (5.5 Mb.)....................	765.000.-
- FUTURE FX30/10 (10.5 Mb.)..................	875.000.-
- FUTURE FX30/20 (20.5 Mb.)..................	1.040.000.-
- SPECTRAVIDEO SV. 318	49.900,-
- SPECTRAVIDEO SV. 328	67.500,-
- SPECTRAVIDEO SV. 728	64.500,-
- GENERAL LBC - 1100 (CPU)	43.500,-
- GENERAL CL 1000 (PRINTER/CASSETTE)	43.900,-
- GENERAL RJ - 101 AMPLIACION DE MEMORIA......	18.800,-

HARDWARE SPECTRAVIDEOS (PERIFERICOS)

- SV-101	QUICK SHOT I JOYSTICK CONTROLLER	2.400,-
- SV-102	QUICK SHOT II JOYSTICK CONTROLLER ...	2.900,-
- SV-103	QUICK SHOT III JOYSTICK CONTROLLER...	4.990,-
- SV-105	TABLETA GRAFICA	16.900,-
- SV-205	CABLE INTERFACE CENTRONICS	3.995,-
- SV-205A	CABLE INTERFACE CENTRONICS	3.995,-
- SV-206	CABLE MONITOR	1.000,-
- SV-602	MINI EXPANDER	4.500,-
- SV-603	COLECOVISION ADAPTADOR DE JUEGOS	12.000,-
- SV-605	EXPANDER, CONTROLADOR DISCO,1 LECTOR DISCO, INTERFACE CENTRONICS	99.900,-
- SV-605A	EXPANDER, CONTROLADOR DISCO, 2 LECTORES DISCOS, INTERFACE CENTRONICS	148.000,-
- SV-605B	EXPANDER CON DOS FLOPPYS 320K DISCO .	176.500,-
- SV-605B	EXPANDER CON DOS FLOPPYS 320K DISCO CON PROGRAMAS.....................	216.500,-
- SV-803	16K RAM MEMORIA	7.900,-
- SV-805	INTERFACE RS 232	15.500,-
- SV-806	TARJETA 80 COLUMNAS	23.800,-
- SV-807	64K RAM MEMORIA	21.900,-
- SV-904	CASSETTE	7.900,-

BROTHER HARDWARE		P.V.P./Ptas.
- HR-1	IMPRESORA DE MARGARITA CENTRONICS.	182.000,-
- HR-1	IMPRESORA DE MARGARITA RS-232	187.000,-
- HR-5	IMPRESORA TERMICA CENTRONICS	39.950,-
- HR-5	IMPRESORA TERMICA RS-232	39.950,-
- HR-15	IMPRESORA DE MARGARITA CENTRONICS.	116.000,-
- HR-15	IMPRESORA DE MARGARITA RS-232	121.000,-
- HR-25	IMPRESORA DE MARGARITA CENTRONICS.	195.100,-
- HR-25	IMPRESORA DE MARGARITA RS-232	199.950,-
- HR-35	IMPRESORA DE MARGARITA	216.000,-
- M1009	IMPRESORA DE MATRIZ CENTRONICS ..	47.500,-
- M1009	IMPRESORA DE MATRIZ Dual	49.950,-
- 2024L	IMPRESORA DE AGUJA Cent	241.500,-
- EP44	MAQUINA DE ESCRIBIR CON INTERFACE.	68.900,-
- HR-1 (T)	ALIMENTADOR TIPO TRACTOR HR-1	19.850,-
- TF-50	ALIMENTADOR TIPO TRACTOR HR-15/25.	19.850,-
- HR-15 (C)	ALIMENTADOR DE HOJAS SUELTAS HR-15	42.500,-
- CF-100	ALIMENTADOR DE HOJAS HR/25/35	48.800,-
- CB-50	TECLADO PARA HR-15	31.950,-

NOTA: Precios sujetos a variacion sin previo aviso.
En el precio no va incluido el I.T.E.

SALUDOS

Desde esta página vamos a comentar algunos temas que nos han sido transmitidos por algunos socios.

SOBRE EL SOPORTE TECNICO

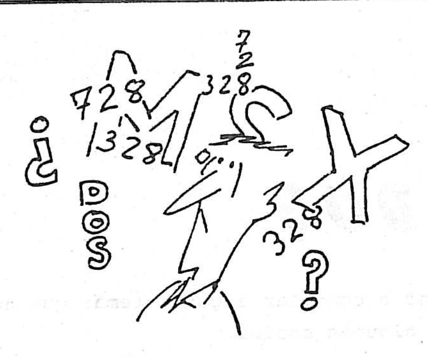
Muchos usuarios se quejan porque su proveedor habitual no tiene los conocimientos tecnicos necesarios para solucionarle algún tipo de duda tanto tecnica como de marca (accesorios, periféricos, posibilidades, etc.) pero si consideramos que hoy por hoy este tipo de Ordenadores se comercializan en establecimientos no necesariamente especializados, la cosa es lo más natural del mundo, pues para ser comercial de informática es necesario tener una base y una preparación muy específica que no se adquiere en cuatro días.

Así mismo el hecho de que el comercio esté super surtido de marcas y accesorios, no quiere decir que los dependientes conozcan sus productos, por ello el servicio "pos venta" suele ser necesariamente deficiente, dado que un Ordenador es bastante diferente de una aspiradora o un televisor, incluso de una emisora para radioaficionado.

Todo esto viene como explicación a muchos socios con los que hemos hablado y con ello queremos dejar claro que la imagen de un Ordenador no es necesariamente la que puede dar el comercio que nos lo vendió.

¿ Para qué estamos los informáticos ?

Los informáticos existimos como tales, con el fin de comercializar, Aplicar y mantener los productos del mismo nombre.



ASUNTO 728 / 328.

Con el lanzamiento del 728 se ha originado una verdadera polémica sobre el SVI más adecuado. ¿ 328 o 728 ?

Vamos a hablar sobre uno y otro, para disipar cualquier duda que pudiese existir al respecto.

El SVI 728 es realmente una consola con la apariencia externa del 328, pero con la diferencia de que lleva implantado el sistema M S X D O S que lo hace totalmente compatible con los programas y periféricos de otras marcas del mismo sistema (Unidades de Diskette, juegos, etc.).

Esto en principio es muy atractivo, pero depende de la aplicación que pensemos darle al sistema.

El 728 , al tener que cumplir las normas MSX de acuerdo con las otras marcas ha perdido la flexibilidad de crecimiento que tenía en un principio y ha ganado en compatibilidad con la familia MSX

Un ejemplo de esto es que al 728 no se le puede acoplar el super expander 605 en ninguna de sus modalidades y como es lógico pierde todas las posibilidades que este periférico brinda.

No Obstante, el 728 es un Ordenador orientado al Hogar y a una informática personal con el techo bastante limitado.

El SVI 328 es como todos conocemos un Ordenador de la primera generación MSX pero con más posibilidades de expansión que podemos dominar inusuales en este tipo de Ordenadores.

En un futuro inmediato (seguramente en el próximo boletín) este modelo asombrará a todos los usuarios por su potencia expansiva.

Por otro lado el 328 no es compatible con el MSX DOS pero con el acoplador adecuado es tan MSX como el 728, de modo que tenemos las dos opciones en un solo Ordenador.

Resumiendo, son dos orientaciones diferentes para cubrir dos tipos de demanda informática.

- La casera y personal, sin grandes pretensiones (728)
- La informática personal con miras a una evolución profesional potenciando el Ordenador de acuerdo con las necesidades, sin que éste se quede pequeño (328)


```
28Ø Print "Ha salido el número"; X
29Ø Rem Inicialización de operaciones.
3ØØ H$= "F"
31Ø If X/2= INT(X/2) Then K$= "P" :GOTO 33Ø
32Ø K$="I"
33Ø If X=Ø then print "Ha ganado la Banca":GOTO 45Ø
34Ø If X=Z then P=P+37*4 : GOTO 36Ø
35Ø I=I+1
36Ø If A$ = K$ then P=P+2*Y1 : GOTO 38Ø
37Ø I=I+1
38Ø If B$= H$ then P=P+2*Y2: GOTO  4ØØ
39Ø I=I+1
4ØØ If I=3 Then print "No has ganado nada": GOTO 45Ø
41Ø L= P-P1
42Ø If L=Ø then 1=3: GOTO 4ØØ
43Ø If L Ø then print "Has perdido";-L: GOTO 45Ø
44Ø Print "Has ganado";L
45Ø Print: Print "Tienes por tanto";P: GOTO 4Ø
46Ø END
```

P R O G R A M A S

Agradecemos la colaboración que se empieza a notar en esta sección y que nos ha permitido aumentar el espacio dedicado a software de interés general.

No obstante, rogamos que los listados nos los mandeís lo más claras posibles, sobre todo en cuanto a signos especiales y variables.

A continuación insertamos un sistema de generación de caracteres que nos ha facilitado nuestro asiduo colaborador IÑAKI CASTILLO y al que aprovechamos para felicitarle por sus importantes colaboraciones.

. ganar no ganaremos, pero jugar si jugaremos y al tiempo APRENDEREMOS

Francisco J. Garcia nos ofrece " R U L E T A "

UTILIZACION DE 1158 BYTES DE MEMORIA

Este programa funciona con las reglas internacionales de la Ruleta.

Se entra con una cantidad al juego, apostando a un número, par o impar y pase o falta. El ordenador halla los beneficios y pérdidas, e imprime los resultados. Al acabar este proceso y mientras haya dinero para jugar el ordenador continua.

Espero que este programa les sea útil, y espero que esté bien copiado, porque contando con que lo esté, es un programa entretenido y apasionante.

- R U L E T A -

```
1Ø  REM RULETA
2Ø  PRINT "Con qué cantidad entras al casino?"
3Ø  Input P
4Ø  P1=P
45  N=RDN(-TIME)
5Ø  PRINT :X=INT(RND(1)*37):I=Ø
6Ø  If P=Ø T hen print " Se acabó el juego,  no tienes dinero": GOTO 460
7Ø  Print·"A qué número apuestas?"
8Ø  Input Z
9Ø  Print "Cuanto dinero?"
1ØØ Input Y
11Ø P=P-Y: If P> Øthen 14Ø
12Ø Print "No puedes apostar más dinero del que tienes"
13Ø P=P+Y1 : GOTO 9Ø
14Ø Print "Apuestas pares o impares (P/I)"
15Ø Input A$
16Ø Print "Cuanto dinero"
17Ø Input Y1
18Ø P=P-Y1 : If P>=Ø then 21Ø
19Ø Print "No puedes apostar más dinero del que tienes"
2ØØ P= P=Y1 : GOTO 16Ø
21Ø Print "Apuesta a pase o falta (P/F)"
22Ø Input B$
23Ø Print "Cuanto dinero"
24Ø Input Y2
25Ø P= P-Y2 : If P>=Ø then 28Ø
26Ø Print "No puedes apostar más dinero del que tienes"
27Ø P=P=Y2 :;If P> =Ø Then 28Ø
```


Hemos tenido una petición general que vamos a tratar de complacer:

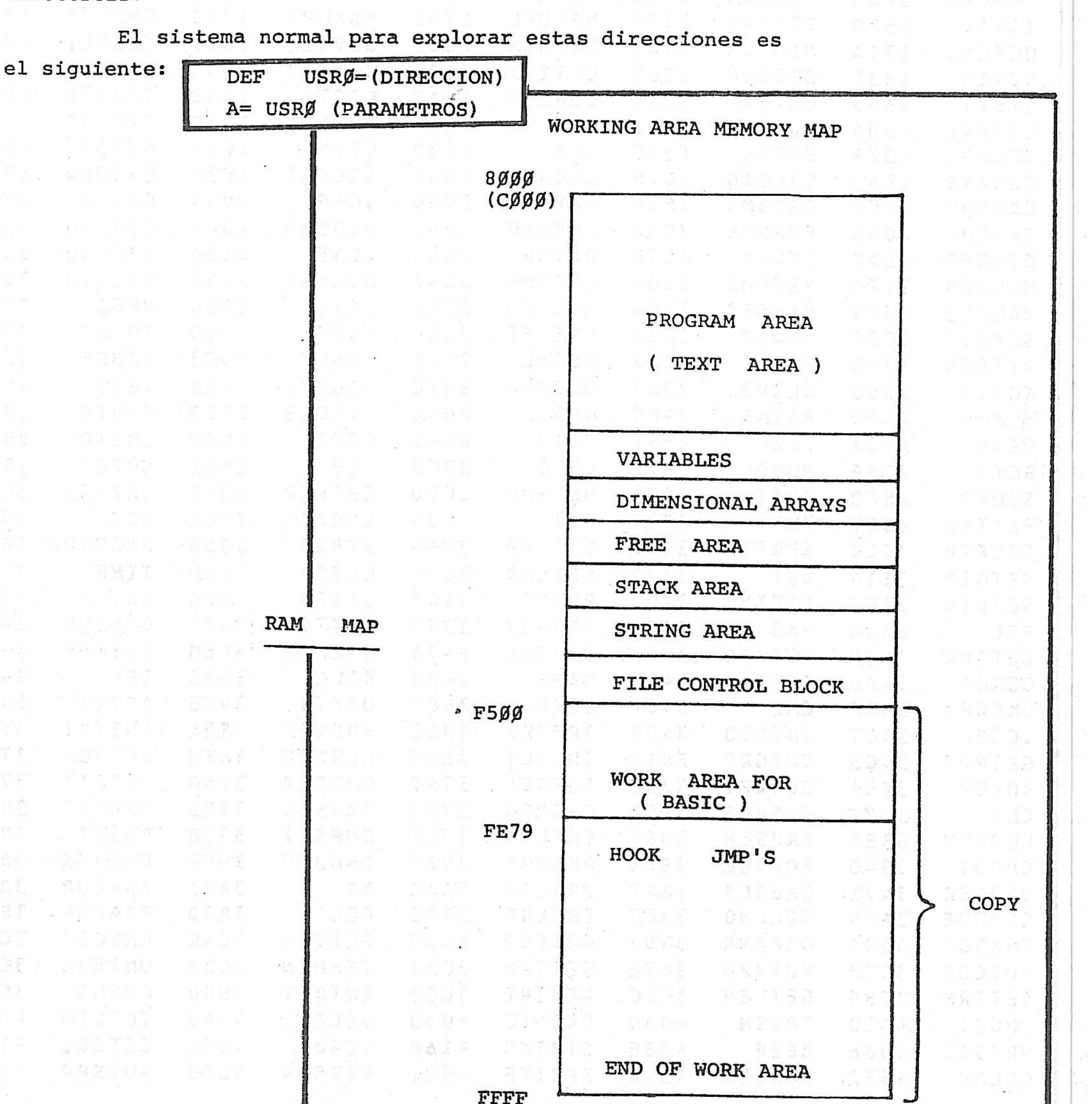
EL MAPA DE MEMORIA

Conseguimos el deseado mapa, pero no logramos una explicación de cada una de las direcciones, utilidad, parámetros etc...

No obstante lo incluímos y esperamos noticias vuestras con el resultado de las investigaciones a que seguramente lo sometereis.

El sistema normal para explorar estas direcciones es el siguiente:

```
DEF   USRØ=(DIRECCION)
A= USRØ (PARAMETROS)
```



OUTOC	0018	FORSZC	0019	SIGN	0029	GETYPR	0030	ORVLEN	004C
FILLEN	0109	IOGOR	01C1	SINFIX	0245	COSFIX	0248	TANFIX	0240
ATNFIX	024F	ALPTAB	0295	OVRMSG	062C	DIVMSG	0684	MLTNAM	0800
TXTCSP	0800	CONSTR	084C	INTXT	089A	REDDY	089F	BRKTXT	08A4
FNOFOR	08AA	READYR	08C9	SNERR	08ED	DVOERR	08F0	NFERR	08F3
ODERR	08F6	OVERR	08FF	MOERR	0902	TMERR	0905	ERROR	0907
ERESET	092E	ERRFIN	0980	STPRDY	09AE	READY	09AF	REPINI	09C1
MAIN	09C4	INILIN	09F8	EDENT	0A28	FINI	0AC9	LINKER	0AE5
CHEAD	0AE9	FNDLIN	0827	CRUNCH	0844	RENCRN	08F3	LBDERR	0027
NOTRFN	002F	NXTCON	0E3A	NEWSTT	0E3E	NWSTRT	0E51	GONE	0E82
CHRGTR	0EA0	CHRGTZ	0EAE	CHRCON	0E82	INTIOX	0F99	INTIO2	0F9A
FCERR	0F9E	LINSPC	0FA3	LINGET	0FA0	GOSUB	0FF6	GOSUB2	1013
GOTO2	1029	USERR	105C	LETCON	1089	NTONG1	1186	LINPT3	12CA
LINPT4	131E	FINPRT	1365	DOASIG	1457	FRMEQL	14C5	FRMPRN	14C8
FRMEVL	14CA	FRMCHK	14C3	TSTOP	1409	EVAL	1620	PARCHK	16E9
ISVAR	16F0	RETVAR	1700	MAKUPL	1708	MAKUPS	170C	CNSGET	1715
OCTCNS	171A	MINPLS	17A9	GRPNAM	1800	GIVDBL	182E	SNGFLT	183C
GIVINT	183E	DOCNVF	19E7	GETINT	1A98	GETIN2	1A99	INTFR2	1A9C
GTBYTC	1AA5	GETBYT	1AA6	CONINT	1AA9	LIST	1A88	SPRATR	1800
LISPRT	1805	BUFLIN	180E	DEL	1C8F	FRMQNT	1C39	FRQINT	1CC3
SCCPTR	1078	DEPTR	1E00	.Ç1	1E13	CSAVE	1E15	CSBSAV	1E3E
CBSAVE	1E43	CBLOAD	1E7F	CLOAD	1EAA	SRCCAS	1F34	CASOPW	1FA7
CASBNW	1FC6	CASBNR	1FE4	GRPCCL	2000	.C48	2014	CASIN	2016
CASOUT	2026	CSRDON	203A	OFFDIO	204A	DIDERR	2040	CWRTON	2059
CTWOFF	206C	CTOFF	207C	DATAW	20E3	.C45	218A	MACLNG	21C0
MCLSCN	21F6	FETCHZ	223C	FETCHR	2242	DECFET	225F	VALSCN	2260
VALSC2	2279	VARGET	22C4	MCLXEQ	22CC	.C26	2200	NEGD	220F
SCAN1	220F	SCAND	22EE	PRESET	2328	PSET	2320	POINT	2346
ATRSCN	2390	XDELT	2331	NEGHL	2388	YDELT	23C3	XCHGY	23CE
XCHGX	2308	GLINE	23E7	DOGRPH	247C	DOGRP2	2488	.C23	24F0
HLFDE	24F4	PAINT	24FC	NEGDE	264C	CIRCLE	2652	GPUTG	28A8
DRAW	290A	.C24	2841	PUTQ	2845	GETQ	2860	INITQ	2880
BCKQ	289E	NUMQ	28AE	LFTQ	28C0	.C9	28E1	MOTOR	28E5
SOUND	28F0	PLAY	2C24	MCLEND	2C00	GETVCP	2046	GETVC1	2040
PLYTAB	2087	PUT	2F81	GET	2F84	LOCATE	2FD1	MDM	3036
STOPTP	303C	SPRTTP	3042	INTTRP	3048	STRIG	3056	ONGOTP	3088
SETGSB	3110	KEY	3120	KEYCHR	3144	CLICK	31AF	TIME	3190
GETLIN	31C7	STTIME	3103	PLAYF	310E	STICK	3206	TRIGF	3263
POL	3290	PAD	3290	RETSWI	3377	SWITCH	337F	CHKBNK	3420
GETBNK	3463	PUTBNK	346A	JMPBNK	3476	CALBNK	3480	RSTFNK	3498
OSKO$	34A6	SETS	34A8	NAME	3480	KILL	3485	IPL	348A
OKCOPY	348F	CMD	34C4	OSKF	34C9	OSKI$	34CE	ATTR$	3403
.C25	3407	INITIO	3409	BREAKX	3512	VDPWRT	3536	INITXT	3541
GETPAT	35C5	INIGRP	3610	INIMLT	3665	CLRSPR	368E	WRTVDP	372A
RDVDP	3734	SETWRT	373C	SETRD	3747	CHGCLR	3750	TOTEXT	3768
CLS	3777	CLSHRS	3798	CHGMOD	3709	SAVSCN	37E6	SPRPAT	3800
LOOSCN	3831	VRFSCN	3887	CHPLPT	3915	CHPSTT	3938	POSIT	393E
CHPUT	3940	POPALL	3966	PSOHRT	3967	INDJMP	399F	CXOPCS	3A6C
OSPCSR	3A71	CKERCS	3AA7	ERACSR	3AAC	BS	3AC1	ADVCUR	3A00
CSHOME	3AF4	DELLNO	3AF0	INSLNO	382C	ECL	3860	ERAFNK	3886
FNKSB	3895	OSPFNK	389F	GETCOD	3C39	GETVRM	3C4C	CNVCOD	3C5A
PUTCOD	3C78	PUTVRM	3C7E	GETTRM	3CA7	TERMIN	3C83	UNTERM	3C84
SETTRM	3C85	GETLEN	3C8C	KEYINT	3CC2	INTRET	3080	CHSNS	30CA
CHGET	4030	TRYIN	4030	CKCNTC	405D	GICINI	4066	IGICIN	4066
WRTGIC	4086	BEEP	408E	STRTMS	416E	.C40	4196	CGTABL	4198
COLOR	4552	SCREEN	459A	SPRITE	4502	RETSPR	4606	PUTSPR	464C

```
3Ø56 IFAL<6ØØANDAL>=5ØØTHEN3Ø85
3Ø58 IFAL<5ØØANDAL>=4ØØTHEN3Ø87
3Ø6Ø IFAL<4ØØANDAL>=3ØØTHEN3Ø89
3Ø62 IFAL<3ØØANDAL>=2ØØTHEN3Ø91
3Ø64 IFAL<2ØØANDAL>=1ØØTHEN31ØØ
3Ø66 IFAL<1ØØANDAL>=5ØTHEN311Ø
3Ø69 Q=66: GOTO 34ØØ
3Ø7Ø Q=78: GOTO 34ØØ
3Ø75 Q=9Ø: GOTO 34ØØ
3Ø8Ø Q=1Ø2: GOTO 34ØØ
3Ø85 Q=114: GOTO 34ØØ
3Ø87 Q=126: GOTO 34ØØ
3Ø89 Q=138: GOTO 34ØØ
3Ø91 Q=15Ø: GOTO 34ØØ
31ØØ Q=162: GOTO 34ØØ
311Ø Q=179: GOTO 34ØØ
34ØØ PSET(15Ø,Q),1Ø
34Ø1 PSET(149,Q),1Ø
34Ø2 PSET(151,Q),1Ø
34Ø3 PSET(151,Q+1),5
34Ø4 PSET(15Ø,Q+1),5
34Ø5 PSET(149,Q+1),5
34Ø6 PSET(15Ø,Q+2),1Ø
34Ø7 PSET(149,Q+2),1Ø
34Ø8 PSET(151,Q+2),1Ø
34Ø9 PSET(15Ø,Q+3),6
341Ø PSET(15Ø,Q+4),6
3411 PSET(15Ø,Q+5),6
3412 PSET(152,Q+1),13
3413 PSET(153,Q+2),13
3414 PSET(154,Q+3),13
3415 PSET(148,Q+1),13
3416 PSET(147,Q+2),13
3417 PSET(146,Q+3),13
3418 PSET(154,Q+4),13
3419 PSET(146,Q+4),13
342Ø PSET(149,Q-1),1Ø
3421 PSET(15Ø,Q-1),1Ø
3422 PSET(151,Q-1),1Ø
3423 J=Q-5
3434 IF J>134 THEN J=134
3436 LINE(145,61)-(155,J),1,BF
3437 IF Q>14Ø THEN LET K=Q-5
3438 IF K<135 THEN K=135
3439 LINE(145,134)-(155,K),14,BF
35ØØ RETURN
```

Francisco Javier de los Dolores Lupion

```
630 PRINT"APARTE DE AGUJEREAR LA LUNA QUE MAS SABES HACER?":PLAY"
    ADEEDF":  GOTO 7ØØ
635 GOTO 7ØØ
64Ø PRINT"QUE NOMBRE LE VAS A PONER AL CRATER?":PLAY"EFFEFDFF":GOTO7ØØ
65Ø PRINT"ESTAS SEGURO DE SABER COMO SE JUEGA":PLAY"O7CDEAB"
7ØØ PRINT
71Ø INPUT"QUIERES SEGUIR ALUNIZANDO(S/N)";IJ$
72Ø IFIJ$="S"ORIJ$="s"THEN GOTO 23Ø
73Ø PRINT:PRINT
74Ø PRINT"CONFIO EN QUE SEPAS REGRESAR A LA       TIERRA"
75Ø END
1ØØØ COLOR 11,1,1:LINE(15,1)-(22Ø,1):LINE(15,1)-(15,35):LINE(45,1)-
     (45,35):LINE(9Ø,1)-(9Ø,35):LINE(165,1)-(165,35):LINE(22Ø,1)-
     (22Ø,35):LINE(15,35)-(22Ø,35)
11ØØ RETURN
2ØØØ LINE(3Ø,6Ø)-(2Ø5,19Ø),6,B
2Ø1Ø A$="BM31,11ØC14R12D1R9D1R9D1R5D1R6D1R5D1R3D1R5D1R3D1R4D1R4D1R3
     D1R3D1R4D1R4D1R4D1R5D1R2D1R4D1R4D1R6D1R5D1R3D1R4D1R3D1R7D1R3D1
     R4D1R3D1R5D1R3D1R4D1R5D1R2D1R4D1R5D1R3D1R3D1R6D41L174U79
2Ø11 DRAW A$
2Ø12 PAINT(32,111),14
2Ø13 CIRCLE (6Ø,165),15,15,,,2/5
2Ø14 CIRCLE(13Ø,15Ø),9,15,,,2/5
2Ø15 CIRCLE(185,8Ø),1Ø,5
2Ø16 PAINT (185,8Ø),5
2Ø17 CIRCLE(18Ø,75),3,1Ø,,,4/5
2Ø18 PAINT(18Ø,75),1Ø
2Ø19 CIRCLE(185,85),4,1Ø,,,3/5
2Ø2Ø PAINT(185,85),1Ø
2Ø21 CIRCLE(187,8Ø),3,1Ø
2Ø22 PAINT (185,8Ø),1Ø
2Ø23 CIRCLE(185,169),1Ø,15,,,2/3
2Ø24 CIRCLE(4Ø,13Ø),4,15,,,1/2
2Ø25 CIRCLE(9Ø,13Ø),7,15,,,1/4
2Ø3Ø PSET(31,11Ø),15
2Ø31 PSET(36,69),15
2Ø32 PSET(48,98),15
2Ø33 PSET(196,1ØØ),15
2Ø34 PSET(195,9Ø),15
2Ø35 PSET(145,7Ø),15
2Ø4Ø PSET(9Ø,12Ø),15
2Ø5Ø PSET(12Ø,11Ø),15
2Ø51 PSET(1Ø1,1ØØ),15
2Ø52 PSET(15Ø,94),15
2Ø53 PSET(12Ø,99),15
2Ø54 PSET(13Ø,75),15
2Ø55 PSET(64,99),15
2Ø56 PSET(84,75),15
2Ø57 PSET(1ØØ,89),15
2Ø58 PSET(12Ø,11Ø),15
2Ø59 PSET(17Ø,129),15
2Ø6Ø PSET(19Ø,138),15
22ØØ RETURN
2419 PSET(146,Q+4),13
242Ø PSET(15Ø,Q+6),6
25ØØ COLOR12,1,1:LOCATE8Ø,18Ø:PRINT"A L E R T":PLAY"o4ao5b"
26ØØ RETURN
3ØØØ IFAL>9ØØTHEN3Ø69
3Ø5Ø IFAL<9ØØANDAL>=8ØØTHEN3Ø7Ø
3Ø52 IFAL<8ØØANDAL>=7ØØTHEN3Ø75
3Ø54 IFAL<7ØØANDAL>=6ØØTHEN3Ø8Ø
```


VPOKE	4609	VPEEK	46F2	GRPPRT	4702	PRLOGO	4782	SCALXY	48A1
CHKMOC	48E1	MAPXYC	48E9	FETCHC	4943	STOREC	494A	READC	4951
SETATR	4980	SETC	4988	RIGHTC	49CF	LEFTC	49F9	TDOWNC	4A14
DOWNC	4A2D	TUPC	4A3F	UPC	4A59	NSETCX	4AF9	GTASPC	48C3
PNTINI	48C9	SCANR	4300	SCANL	4C66	PIXSIZ	4CC9	PGINIT	4CCF
NREAD	4001	NWRITE	4021	.C2	4081	DECSUB	4086	DSUB	4086
DADDS	4091	DADD	4094	DECADD	4094	DECNRM	40F6	DECROU	4E38
DECRC3	4E30	DECRDA	4E44	DECSRD	4EF3	DECMUL	4EFE	DMULT	4EFE
DECMRN	4F46	DDIV	4F87	DECDIV	4F87	.C20	5066	COS	5098
SIN	5001	TAN	5120	ATN	5139	LOG	5197	SQR	5222
EXP	5268	RND	5300	RNDINI	5343	RNDMN2	534C	DMULTO	5364
MAF	5376	MAM	5379	MFA	5382	MFM	5385	MMA	538A
MMF	538F	XTF	5397	PHA	53F1	PHF	53F6	PPA	5406
PPF	540C	.C21	540D	DBLZER	543E	ONE	5446	SIGNC	55A1
SIGNS	55A8	INRART	55AA	ZERO	55AC	ABSFN	5581	VNEG	5585
NEG	558C	SGN	55C6	CONIA	55C9	VSIGN	55D0	ISIGN	55DA
PUSHF	55E0	MOVFM	55ED	MOVFR	55F0	MOVRF	55F8	MOVRMI	5605
MOVRM	560E	GETBCD	5610	INXHRT	5615	MOVMF	5617	MOVE	561A
VMOVAM	561E	VMOVE	5622	MOVE1	5626	MOVE1R	562D	VMOVFA	5634
VMOVFM	5637	VMOVAF	563C	VMOVMF	563F	VDFACS	5643	FCOMP	5650
ICOMP	567A	XDCOMP	5689	DCOMP	56AE	FRCINT	5685	CONIS	568D
MAKINT	56C4	VALINT	56C7	CONIS2	56CD	FRCSNG	56D0	CONSD	56E5
CONSI	56F3	CONSIH	56F6	FRCDBL	5765	CONDS	576D	VALDBL	577A
VALSNG	577E	CHKSTR	5783	FRCSTR	5783	QINTA	5788	DCXBRT	57E7
FIXER	57E9	VINT	57F8	INT	5804	UMULT	5873	ISUB	5890
IADD	5898	IMULT	589C	IDIV	590F	INEGHL	5953	INEG	5950
INEG2	5968	IMOD	596C	FADDS	597D	FADD	5980	FSUB	5989
FMULT	598E	FDIVT	5997	FDIV	5999	CONASD	5982	.C19	59C2
DCRART	59C5	DCXHRT	59C7	POPHRT	59C9	FIN	59C3	FINDBL	59C3
.C18	581A	INPRT	583C	LINPRT	5844	LINOUT	5848	FOUT	5857
PUFOUT	5858	FOUTB	5E48	FOUTD	5E4C	FOUTH	5E50	NUMLEN	5E90
.C17	5EE4	SNGEXP	5EF6	DBLEXP	5F05	INTEXP	5F6D	.C22	605A
DIM	6061	PTRGET	6066	PTRGT2	6068	NOARYS	60CA	PTRGTN	611F
PTRGTR	6122	ERSFIN	61AA	BSERR	610F	.C14	6270	PRINUS	6273
.C13	6400	OUTCON	6407	OUTDLP	6415	LPTCHR	643A	FINLPT	6430
PRINTW	6446	TTYCHR	6455	LINPT1	6456	CRDONZ	6463	FININL	646A
CRDO	6474	CRFINC	647C	CRFIN	647D	ISCNTC	6495	CKSTTP	640F
INKEY	64F3	OUTCH1	6513	.C10	6519	BLTU	6520	BLTUC	6523
GETSTK	652E	REASON	6537	OMERR	6545	OMERRR	6550	SCRATH	6556
SCRTCH	6557	RUNC	656A	CLEARC	6571	CLEARO	6577	STKINI	6585
STKERR	65C0	GTMPRT	65E7	ONTRP	65EB	OFFTRP	65FB	STPTRP	6601
RSTTRP	660E	REQTRP	6618	SETTRP	6633	FRETRP	6630	INITRP	6653
GOTRP	666E	RESTOR	66AE	RESFIN	66C3	STOPP	66C8	STOP	66C3
STOPRG	66CC	ENDST	66CF	CONSTP	6609	STPEND	66E3	ENDCON	66E6
CTROPT	6709	CTRLPT	6708	CONT	6719	TON	672F	TOFF	6730
SWAP	6735	ERASE	676E	POPAHT	6798	ISLET	679E	ISLET2	679F
CLEAR	67A6	SUBDE	681A	NEXT	6821	ISFLIO	68C2	.C15	68C7
STRCMP	68CD	STROS	68FA	STRHS	68FF	STR8S	6904	STRS	6909
STRCPY	6916	STRIN1	692A	STRINI	692C	STRAD1	6932	PUTDEI	6934
STRLIT	693A	STRLTI	6938	STRLT3	6930	STRLT2	693E	PUTNEW	6959
PUTTMP	6950	STROUI	697C	STROUT	6970	STRPRT	6980	GETSPA	6993
PPSWRT	69AC	GARBAZ	6988	CAT	6A8C	FRESTR	6A05	FREFAC	6A08
FRETM2	6A08	FRETMP	6A0C	FRETMS	6AF3	LEN	6804	ASC	6810
ASC2	6814	CHR$	6820	SETSTR	6826	FINBCK	682A	STRNG$	682E
SPACE$	6840	LEFT$	6866	LEFTUS	6860	RIGHT$	6896	MID$	689F
VAL	68C0	INSTR	68F0	LHSMID	6C73	FRE	6CF7	.C12	6011

PINLIN	6013	QINLIN	6026	INLIN	602F	.C33	6F01	NAMSCN	6F03
NAMSC1	6F06	SCNBLK	702F	GETFLP	7033	GETPTR	7036	FILSCN	7067
SETFIL	7073	OPEN	7080	NULOPN	70C3	CLSFIL	70EA	NOCLSB	710A
LRUN	711F	LOAD	7121	MERGE	7122	SAVE	7167	SPSVEX	71A8
CHKTOP	7209	NOROOM	7210	GETDEV	721A	RSET	7227	LSET	7228
FIELD	72CD	MKI$	7312	MKS$	7315	MKD$	7318	CVI	7328
CVS	732E	CVD	7331	CLOSE	7375	CLSALL	737D	CLSCLR	7388
LFILES	73AD	FILES	7382	DPUT	7388	DGET	7389	FILOUT	73C9
FILOU1	73CA	INDSKC	73F1	INDSKE	7402	FIXINP	7406	CLRBUF	7469
DOCLR	7474	GETBUF	747A	GETBF1	747D	LOC	7484	LOF	749A
EOF	7480	FPOS	74C6	DIROG	74D9	FILINP	74E6	FILGET	74E9
PRGFIN	750C	FILIND	7514	DLINE	7520	NOSKCR	7500	DERBFN	75FA
DERFAO	75FD	DERFOR	7600	DERFNF	7603	DERFNO	7606	DERFOV	7609
DERIFN	760C	DERIER	760F	DERRPE	7612	DERSAP	7615	DERSOO	7618
BSAVE	7624	BLOAD	7684	CHKBRN	76EA	.C6	7706	PARDEV	7708
DEVTBL	7788	GENDSP	77A8	.C3	779A	KBDOSP	77CC	CRTOSP	7817
CASDSP	7841	MDMOSP	791F	LPTDSP	7953	.C29	7989	CHKMOM	798E
DIAL	79C2	SCMTRP	79C8	RCVX	7903	RS2INT	7908	.C28	7909
BOOT	79CC	.C47	7A53	INIDAT	7A66	FNKROM	7A84	.C30	793E
MON	7B44	MONERR	7B4A	.C27	784E	INIT	7850	INIENT	789F
SETMAX	7CBA	DEFILE	7CDA	.C5	7044	LASTWR	7080	C000	.C000
INITSA	F500	RAMLOW	F500	RNOCNT	F504	RNDTAB	F506	USRTAB	F528
ERRFLG	F53F	LPTLST	F540	LPTPGS	F541	PRTFLG	F542	LINLEN	F543
RUBSW	F545	STKTOP	F546	CURLIN	F548	TXTTAB	F54A	VLZADR	F54C
KBUF	F54F	BUFMIN	F68D	BUF	F68E	ENDBUF	F790	TTYPOS	F791
DIMFLG	F792	VALTYP	F793	CONSAV	F798	MEMSIZ	F7A2	TEMPPT	F7A4
TEMPST	F7A6	DSCTMP	F7C4	DSCPTR	F7C5	FRETOP	F7C7	TEMP3	F7C9
TEMP8	F7CB	ENDFOR	F7CD	SUBFLG	F7D1	USFLG	F7D2	TEMP	F7D3
PTRFLG	F7D5	AUTFLG	F7D6	SAVTXT	F7D8	SAVSTK	F7DD	ERRLIN	F7DF
DOT	F7E1	ONELIN	F7E5	ONEFLG	F7E7	TEMP2	F7E8	OLDLIN	F7EA
OLDTXT	F7EC	VARTAB	F7EE	ARYTAB	F7F0	STREND	F7F2	DATPTR	F7F4
DEFTBL	F7F6	PRMSTK	F810	PRMLEN	F812	PARM1	F814	PRMPRV	F878
PRMLN2	F87A	PRMFLG	F8E0	ARYTA2	F8E1	NOFUNS	F8E3	TEMP9	F8E4
FUNACT	F8E6	VLZDAT	F8E8	SWPTMP	F8E9	TRCFLG	F8F1	FBUFFR	F8F2
FMLTT1	F91A	FMLTT2	F918	DECTMP	F910	DECTM2	F91F	DECCNT	F921
DAC	F923	FAC	F923	FACLO	F925	HOLD8	F933	HOLD5	F948
HOLD2	F963	HOLD	F968	ARG	F974	RNDX	F984	MAXORV	F98C
MAXFIL	F98D	FILTAB	F98E	ORVTAB	F990	NULBUF	F992	CURORV	F994
ORVPTR	F995	PTRFIL	F997	FREPLC	F999	LSTFRE	F998	FILMOC	F990
RUNFLG	F990	FILNAM	F99E	FILNM2	F9A7	LSTTRK	F9B0	LSTSCT	F9B1
NLONLY	F9B2	SAVFLG	F9B3	SAVEND	F9B4	DSKBSY	F986	ERRCNT	F9B7
ERRCN1	F9B8	RAWFLG	F989	EBCFLG	F98A	SAVEBC	F988	STATO	F98C
STAT1	F9B0	TSTACK	F98E	FRSTIO	FA00	INIRAM	FA00	CLIKSW	FA02
CSRY	FA03	CSRX	FA04	CSRSW	FA05	CNSDFG	FA06	RG1-SAV	FA07
TRGFLG	FA08	SPCFLG	FA09	FORCLR	FA0A	BAKCLR	FA08	BORCLR	FA0C
MAXUPD	FA0D	MINUPD	FA10	ATRBYT	FA13	PUTFN	FA14	QUEUES	FA17
REPCNT	FA19	PUTPNT	FA1A	GETPNT	FA1C	FNKSTR	FA1E	XOOFLG	FA8E
COMMSK	FA8F	CHKROM	FAC0	CLOC	FAC0	CMASK	FAC2	ASPECT	FAC3
CENCNT	FAC5	CLINEF	FAC7	CNPNTS	FAC8	CPLOTF	FACA	CPCNT	FACB
CPCNT8	FACD	CRCSUM	FACF	CSTCNT	FAD1	CSCLXY	FAD3	CSAVEA	FAD4
CSAVEM	FAD6	CXOFF	FAD7	CYOFF	FAD9	LOHMSK	FADB	LOHDIR	FADC
LOHADR	FADD	LOHCNT	FADF	SKPCNT	FAE1	MOVCNT	FAE3	POIREC	FAE5
LFPROG	FAE6	RTPROG	FAE7	PUTFLG	FAE8	MINDEL	FAE9	MAXDEL	FAEB
ARYPTR	FAED	MCLTAB	FAEF	MCLFLG	FAF1	QUETAB	FAF2	QUEBAK	FB0A
VOICAQ	FB0E	VOICBQ	FB8E	VOICCQ	FC0E	RS2IQ	FC8E	PRSCNT	FCCE
SAVSP	FCCF	VOICEN	FCD1	SAVVOL	FCD2	MCLLEN	FCD4	MCLPTR	FCD5

..... *Y SEGUIMOS con* **SOFTWARE** *LISTAMOS uno de JUEGOS*

"ALUNIZAJE"

Por Francisco de los Dolores Lupión

```
10  COLOR 4, 10, 10
50  SCREEN2:LOCATE25,6Ø:PRINT"alunizaje"
55  FORZ=1TO5ØØ:NEXT
57  PLAY"O1CDB","O6defaA"
11Ø SCREEN1:
115 COLOR 1,1Ø,1Ø
13Ø LOCATE1Ø,25:PRINT"Para frenar la nave en su descenso, has de pulsar
    las teclas 1 al 9 para dar gas según se vaya desarrollando el juego"
14Ø LOCATE1Ø,5Ø:PRINT"Si no pulsas ningún número se te acaba el combus-
    tible, la nave se irá acelerando."
15Ø LOCATE1Ø,7Ø:PRINT"Trata de llegar al suelo (altura Ø) con la menor
    velocidad posible."
16Ø LOCATE6Ø,1ØØ:PRINT"¡¡¡S U E R T E¡¡¡"
17Ø LOCATE4Ø,125:PRINT"CUANDO ESTES PREPARADO PULSA '1'"
2ØØ REM COMIENZO DEL JUEGO
22Ø I$=INKEY$:IFI$<>"1"THEN22Ø
23Ø TP=Ø:GS=Ø
24Ø AL=6ØØ+INT(RND(1)*6ØØ)
25Ø VL=3Ø+INT(RND(1)*3Ø)
26Ø FL=4Ø+INT(AL/8)
27Ø COLOR Ø,Ø,Ø
3ØØ REM CALCULO Y PRESENTACION DE PARAMETROS
31Ø SCREEN 1
32Ø GOSUB1ØØØ
33Ø GOSUB2ØØØ
34Ø COLOR 3,1,1:LOCATE2Ø,4:PRINT"FUEL":LOCATE6Ø,4:PRINT"GAS":LOCATE95,4:
    PRINT"VELOCIDAD":LOCATE17Ø,4:PRINT"ALTURA"
345 LINE(16,24)-(39,32),1,BF:LINE(63,24)-(75,32),1,BF:LINE(1ØØ,24)-
    (14Ø,32),1,BF:LINE(17Ø,24)-(21Ø,32),1,BF:LINE(4Ø,5Ø)-(6Ø,58),1,BF:
    LINE(Ø,58)-(23,17Ø),1,BF
35Ø COLOR 9,1,1: LOCATE15,25:PRINTFL:LOCATE6Ø,25:PRINTGS:LOCATE11Ø,25:
    PRINTVL:CATE175,25:PRINTAL
36Ø PRINT
37Ø IFFL>ØTHEN LOCATE15,5Ø:PRINT"GAS";
38Ø IFFL=Ø THENLOCATE9Ø,5Ø:PRINT"AGOTADO EL COMBUSTIBLE":PLAY"o7ab"
39Ø GS=Ø
4ØØ TP=TP+1
41Ø FORTR=ØTO25Ø
42Ø I$=INKEY$:IFI$<>""THENGS=VAL(I$):PRINTGS
43Ø NEXT
44Ø IFFL<GS THENGS=FL
45Ø FL=FL-GS
46Ø VL=VL+4-GS
47Ø AL=AL-VL:IFAL<1ØØANDVL>1ØTHENGOSUB25ØØ
475 GOSUB3ØØØ
48Ø IF AL>Ø GOTO 34Ø
5ØØ SCREENØ
51Ø CLS
515 COLOR 1Ø,1,1
52Ø PRINT"ALUNIZAJE COMPLETADO":PLAY"O2ge"
53Ø PRINT:PRINT"HAS TARDADO":TP*2:"segundos"
54Ø PRINT:PRINT"LA VELOCIDAD DE LLEGADA ES DE";VL*36;"Km/h."
545 PRINT
55Ø IFVL>2ØTHENPRINT"HAS HECHO UN CRATER DE";VL/2;"Metros."
56Ø RS=INT(VL/1Ø)+1:IFRS>6 THENRS=6
565 PRINT:PRINT
57Ø ONRSGOTO6ØØ,61Ø,62Ø,63Ø,64Ø,65Ø
6ØØ PRINT"ENHORABUENA,ALUNIZAJE EXCELENTE":COLORS,15,15:PLAY"O7CDAC"
6Ø5 GOTO7ØØ
61Ø PRINT"UN POCO BRUSCO,PERO NO HA ESTADO MAL":PLAY"ACDE": GOTO 7ØØ
62Ø COLOR 1,11,11:PRINT"TE HAS CARGADO LA NAVE PERO SIGUES VIVO":PLAY"
    T1ØØO6EFFEEDFF":PLAY"T1ØØO6EEDFFEEFFE":GOTO7ØØ
```

```
10530 LET L2=0
10600 LET J2=J2-365
10700 GOTO 10000
11000 LET M2=I-1
11100 LET D6=J2-J4
11150 IF J2=60 THEN LET D6=D6+L2
11200 LET D5=Y2*10000+(M2*100)+D6
11300 RETURN
12000 FOR I=1 TO 74
12100 LPRINT"-";
12200 NEXT I
12250 LPRINT
12300 LPRINT": ESTUDIO DE LAS CURVAS BIO-RITMICAS                 ";
12400 GOSUB 13600
12500 LPRINT": NOMBRE , ";A$;
12600 GOSUB 13600
12605 DI$=STR$(D9):DI$=MID$(DI$,6,2)+"/"+MID$(DI$,4,2)+"/"+MID$(DI$,2,2)
12700 LPRINT ": DIA EN ESTUDIO-";DI$;"  DURACION-";J5;" DIAS";
12800 GOSUB 13600
12810 FOR I=1 TO 74:LPRINT"-";:NEXT I:LPRINT
12820 LPRINT":           F=FISICO          E=EMOCIONAL          M=MENTAL ";
12830 LPRINT TAB(74)":"
13200 FOR I=1 TO 74:LPRINT"-";:NEXT I
13210 LPRINT
13250 LPRINT":           BAJO            :            ALTO         :";
13260 LPRINT"    DIA    :CRITICO ";
13400 LPRINT":"
13500 FOR I=1 TO 74
13510 LPRINT"-";
13520 NEXT I
13530 LPRINT
13540 RETURN
13600 LET J=75-LPOS(X)
13700 FOR I=1 TO J-1
13800 LPRINT" ";
13900 NEXT I
14000 LPRINT":"
14100 RETURN
14500 FOR I=1 TO 74:LPRINT"-";:NEXT I:LPRINT
14600 LPRINT": ALTO          DIAS DE GRAN VITALIDAD, EFICIENCIA ";
14610 LPRINT"Y ALTA ENERGIA     ";
14700 GOSUB 13600
14800 LPRINT": BAJO          DIAS DE REDUCIDA EFICIENCIA Y RECUP";
14810 LPRINT"ERACION                ";
14900 GOSUB 13600
15000 LPRINT": CRITICO       DIAS DE SITUACION INESTABLE          ";
15010 LPRINT"                       ";
15100 GOSUB 13600
15150 FOR I=1 TO 74:LPRINT"-";:NEXT I:LPRINT
15200 LPRINT TAB(31);"PAGE ";P1
15400 RETURN
```

QUEUEN	FC07	MUSICF	FC08	PLYCNT	FC09	VC3A	FC0A	VC3B	FCFF
VC3C	FD24	MOMFLG	FD49	STPCPT	FD4A	FRCNEW	FD4B	POLRTY	FD4C
LINTTB	FD4D	FSTPOS	FD65	CODSAV	FD67	FNKSWI	FD68	FNKFLG	FD69
ONGSBF	FD73	CLIKFL	FD74	CLOKEY	FD75	NEWKEY	FD80	SFTKEY	FD86
KEYBUF	FD89	BUFEND	FD83	LINWRK	FD94	PATWRK	FDDC	BOTTOM	FDE4
HIMEM	FDE6	TXPSAV	FDE9	CASATR	FDEA	TRPTBL	FDEB	RTYCNT	FE2A
INTFLG	FE2B	PADY	FE2C	PADX	FE2D	JIFFY	FE2E	INTVAL	FE30
INTCNT	FE32	ESCCNT	FE34	REVFLG	FE35	INSFLG	FE36	CSTYLE	FE37
CAPST	FE38	FLBMEM	FE39	SCRMOD	FE3A	SPRSIZ	FE3B	RGOSAV	FE3C
STATFL	FE3D	KBDPRV	FE3E	CASPRV	FE3F	MOMPRV	FE40	BRDATR	FE41
GXPOS	FE42	GYPOS	FE44	GRPACX	FE46	GRPACY	FE48	DRWFLG	FE4A
DRWSCL	FE4B	DRWANG	FE4C	DATCNT	FE4D	SIOFLG	FE51	RCVXOF	FE52
SNTXOF	FE53	RCVSFT	FE54	SNOSFT	FE55	ADDPM	FE56	RUNBNF	FE57
SAVENT	FE58	REGPC	FE5A	REGSP	FE5C	REGHL	FE5E	REGDE	FE60
REGBC	FE62	REGA	FE64	REGF	FE65	REGFT	FE71	MONFLG	FE72
SAVESP	FE73	SWIFLG	FE75	SPSAVE	FE76	SCNCNT	FE78	HOKJMP	FE79
H.KEYI	FE79	H.PRTF	FE7C	H.DGET	FE7F	H.INOS	FE82	H.SCNE	FE85
H.SNGF	FE88	H.FPOS	FE8B	H.READ	FE8E	H.ISRE	FE91	H.MAIN	FE94
H.RSLF	FE97	H.LOC	FE9A	H.BAKU	FE9D	H.STKE	FEA0	H.PARD	FEA3
H.FRET	FEA6	H.NTFL	FEA9	H.NTFN	FEAC	H.CLEA	FEAF	H.SAVD	FEB2
H.SAVE	FEB5	H.FILE	FEB8	H.LOF	FEBB	H.NTPL	FEBE	H.NODE	FEC1
H.DOGR	FEC4	H.MERG	FEC7	H.EOF	FECA	H.PTRG	FECD	H.NOFO	FED0
H.PRGE	FED3	H.BUFL	FED6	H.CRDO	FED9	H.DKNO	FEDC	H.GETP	FEDF
H.COPO	FEE2	H.DEVN	FEE5	H.SETF	FEE8	H.NULO	FEEB	H.RETU	FEEE
H.CLRC	FEF1	H.LIST	FEF4	H.RUNC	FEF7	H.EVAL	FEFA	H.ISMI	FEFD
H.COMP	FF00	H.FRQI	FF03	H.DIRD	FF06	H.OUTD	FF09	H.NOTR	FF0C
H.GEND	FF0F	H.FILO	FF12	H.ISFL	FF15	H.ERRP	FF18	H.ERRF	FF1B
H.TRMN	FF1E	H.CRUS	FF21	H.CRUN	FF24	H.FINP	FF27	H.FRME	FF2A
H.BINS	FF2D	H.FINI	FF30	H.BINL	FF33	H.FINE	FF36	H.FING	FF39
H.INCH	FF3C	H.WIDT	FF3F	H.PINL	FF42	H.QINL	FF45	H.INLI	FF48
H.DSKC	FF4B	H.ERAF	FF4E	H.OSPF	FF51	H.NEWS	FF54	H.GONE	FF57
H.DMOT	FF5A	H.MOMO	FF5D	H.MOMC	FF60	H.MOMW	FF63	H.MOMI	FF66
H.MOME	FF69	H.MOMB	FF6C	H.DIAL	FF6F	H.RS2I	FF72)	H.DNGD	FF75
H.KYCL	FF78	H.KYEA	FF7B	H.NMI	FF7E	H.KEYC	FF81	H.MON	FF84
H.BAOC	FF87	H.DSKO	FF8A	H.SETS	FF8D	H.NAME	FF90	H.KILL	FF93
H.IPL	FF96	H.COPY	FF99	H.CMD	FF9C	H.DSKF	FF9F	H.DSKI	FFA2
H.ATTR	FFA5	H.MONE	FFA8	H.INIP	FFAB	H.CHPU	FFAE	H.TOTE	FFB1
ENDWRK	FFB4								

En el programa SVWALL hay una línea que está prevista de forma que nunca salen más de tres tandas de ladrillos aunque se juegue todo lo bien que se quiera : Para solucionar esto hay que modificar la línea 470 para que sea como la siguiente :

```
470 P1=P + P2*P: P2=P2 + 1
```

y en la línea 160 donde pone P1 poner simplemente P. Así salen tantos muros nuevos como el jugador sea capaz de soportar.

Una última cosa. En el número 1 del boletín ponía además que las instrucciones DEFUSRØ=Ø:PRINT USR(Ø) inicializan completamente el ordenador; esto no es del todo cierto porque resulta que la RAM comprendida desde el final del BASIC hasta el final no se borra. Esto es probablemente debido a que la RAM original del SPECTRAVIDEO 328 es estática y no se borra mientras no cortemos la alimentación.

GOTITAS DEL SABER

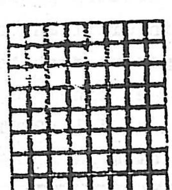
SPECTRAVIDEO SPRITES

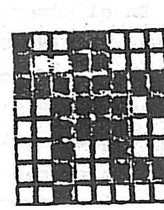
Cuando hablamos de la utilización de SPRITES en el SPECTRAVIDEO, primeramente debemos revisar el método usado en el manual, y entonces añadir detalles que no están claramente detallados o no se encuentran mencionados.

Los SPRITES son imágenes gráficas dinámicas cuyas características de forma, tamaño y color estan determinadas por el programador. Se pueden diseñar de manera que tengan movilidad en la pantalla sin alterar el fondo o contenido de la pantalla. Pueden pasar en frente o detrás de otros sprites, o ser programados para activar una subrutina si se tocan entre ellos.

El SPECTRAVIDEO SVI-328 y 318 le permite a usted tener 32 sprites en la pantalla simultáneamente. Esto es cuatro veces mayor que en el COMMODORE 64, y TEXAS INSTRUMENTS TI-99/4A. No existe en el mercado una unidad que posea mayor capacidad de Sprites incorporadas que el SPECTRAVIDEO. Si usted es una persona que está adentrada dentro del campo de gráficas en los ordenadores, sería un crimen el no tomar ventaja de esto. Su utilidad no se encuentra limitada solo a juegos.

El primer paso en crear un sprite es el diseñar su patrón. Este proceso se comienza dibujando una caja dividida en 8 bloques cada una. (Cada bloque representa un pixel en la pantalla).El patrón es definido llevando los bloques correspondientes. La ilustración siguiente nos da una explicación de este procedimiento.





```
4900 LET J2=J2+D2
5000 LET L1=(Y1/4)-(INT(Y1/4))
5100 IF L1=0 THEN LET L1=1:GOTO 5300
5200 LET L1=0
5300 LET L2=(Y2/4)-(INT(Y2/4))
5400 IF L2=0 THEN LET L2=1:GOTO 5600
5500 LET L2=0
5600 IF M1>2 THEN LET J1=J1+L1
5700 IF M2>2 THEN LET J2=J2+L2
5800 LET D4=D4+J2-J1
6000 LET D1=(D4-(INT(D4/33)*33))
6100 LET D2=(D4-(INT(D4/28)*28))
6200 LET D3=(D4-(INT(D4/23)*23))
6300 FOR L3=1 TO 50
6350 FOR I= 1 TO 50
6360 LET L$(I)="-"
6370 NEXT I
6400 LET X=SIN(R1*D1)
6500 LET Y=SIN(R2*D2)
6600 LET Z=SIN(R3*D3)
6700 LET L$(X*20+25)="M"
6800 LET L$(Y*20+25)="E"
6900 LET L$(Z*20+25)="F"
6950 LPRINT":  ";
7000 FOR I=1 TO 50
7050 LET L$(25)="!"
7100 LPRINT L$(I);
7200 NEXT I
7205 LPRINT": ";
7207 GOSUB 10000:DIA$=STR$(D5):LPRINT MID$(DIA$,6,2);"/";MID$(DIA$,4,2);"/";MID$
(DIA$,2,2);" : ";
7210 IF D1=0 THEN LET C=1:LPRINT"M ";
7215 IF D1=16 THEN LET C=1:LPRINT"M ";
7220 IF D2=0 THEN LET C=1:LPRINT"E ";
7225 IF D2=14 THEN LET C=1:LPRINT"E ";
7230 IF D3=0 THEN LET C=1:LPRINT"F ";
7235 IF D3=12 THEN LET C=1:LPRINT"F ";
7240 IF C=1 THEN LET C=0
7250 GOSUB 13600
7300 LET D1=D1+1
7400 LET D2=D2+1
7500 LET D3=D3+1
7600 IF D1=33 THEN LET D1=0
7700 IF D2=28 THEN LET D2=0
7800 IF D3=23 THEN LET D3=0
7900 LET J2=J2+1
7920 LET J6=J6+1
7950 IF J5<J6 GOTO 8300
8000 NEXT L3
8050 LET P1=P1+1
8100 GOSUB 14500
8125 LPRINT:LPRINT
8150 GOSUB 12000
8200 GOTO 6300
8300 LET P1=P1+1
8350 GOSUB 14500
8400 FOR N=1 TO 17:LPRINT:NEXT N
8500 GOTO 75
10000 RESTORE
10100 FOR I =1 TO 13
10150 LET J4=J3
10200 READ J3
10250 IF J2>59 THEN LET J3=J3+L2
10300 IF J2<=J3 GOTO 11000
10400 NEXT I
10500 LET Y2=Y2+1
10510 LET L2=(Y2/4)-(INT(Y2/4))
10520 IF L2=0 THEN LET L2=1:GOTO 10600
```

Biorritmos

Este es seguramente, uno de los primeros programas de entretenimiento que se implementó en Ordenadores, y aún sigue estando en vigencia por la curiosidad que despierta.

Todos los cálculos realizados son reales y se supone que los resultados son verídicos.

Este programa escribe en impresora los resultados del Biorritmo; para que aparezca en pantalla debe de cambiarse las LPRINT por PRINT.

```
1 LET R1=(360/33)/57.2958
2 LET R2=(360/28)/57.2958
3 LET R3=(360/23)/57.2958
4 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
10 PRINT "******************** BIO-RITMO *********************"
15 PRINT:PRINT
20 PRINT "PARA GENERAR UN BIO-RITMO:"
25 PRINT
30 PRINT "     - ESCRIBA LA FECHA DE NACIMIENTO Y EL DIA DE HOY"
35 PRINT "        (SEPARADOS POR UNA COMA EJ: AAMMDD,AAMMDD)"
40 PRINT "     - INTRODUZCA EL NUMERO DE DIAS A ANALIZAR"
45 PRINT "     - INTRODUZCA EL NOMBRE DE LA PERSONA"
47 PRINT:PRINT:PRINT:PRINT
50 DATA 0,31,59,90,120,151,181,212,243,273,304,334
51 DATA 365
60 DIM L$(50)
70 GOTO 90
75 RUN 'COMIENZA EJECUCION
90 RESTORE
100 PRINT" INTRODUZCA LA FECHA DE NACIMIENTO, EL DIA DE HOY (AAMMDD)"
125 LET P1=0
150 LET J6=1
200 INPUT D1,D2
205 LET D9=D2
206 J5=40:GOTO 210:' MODIFICACIONPRINT" INTRODUZCA LA DURACION "
207 INPUT J5
210 PRINT" INTRODUZCA EL NOMBRE "
220 INPUT A$
230 GOSUB 12000
300 IF D1>D2 THEN PRINT" DATOS ERRONEOS "
400 LET X1=D1
500 GOSUB 1000
550 LET Y1=X2:LET M1=X3:LET D1=X4
600 LET X1=D2
625 GOSUB 1000
650 LET Y2=X2:LET M2=X3:LET D2=X4
800 GOSUB 4000
1000 LET X2=INT(X1/10000)
1100 LET X3=INT(X1/100)-(X2*100)
1200 LET X4=X1-((X3*100)+(X2*10000))
1300 RETURN
4000 LET D4=(INT((Y2-1)*365.25)-INT((Y1-1)*365.25))
4100 FOR I=1 TO M1
4200 READ J1
4300 NEXT I
4400 RESTORE
4500 FOR I =1 TO M2
4600 READ J2
4700 NEXT I
4800 LET J1=J1+D1
```


El próximo paso es convertir esta imagen en data que su ordenador pueda usar para el ensemblaje del sprite. La data con la que se puede trabajar hasta el presente, son el sistema de números binarios. Tomamos una fila de 8 bloques, de arriba hacia abajo (una columna) y considerar un bloque lleno como 1 y cada bloque vacío como cero. Esto producirá 8 números binarios de 8 digitos cada uno. Estos a su vez son introducidos en las lineas de datos. Usted podrá ponerlos todos dentro de úna línea de datos, pero al poner cada valor dentro de su linea correspondiente, Vd. podrá ver el diseño de los Sprites cuando Vd LISTE (LIST) su programa. Claro que Vd. tiene la opción de poner sus datos en los ficheros de su cassette o disco.

Ahora puede Vd. usar el programa FOR NEXT LOOP que se encuentra en su manual con el propósito de ensamblar el sprite y ponerlo en la pantalla.

1	0	0	1	1	0	0	0
1	0	0	1	1	0	0	0
1	1	1	1	1	1	1	1
0	0	1	1	1	1	0	1
0	0	1	1	1	1	0	0
0	0	1	0	0	1	0	0
0	0	1	0	0	1	1	1
0	0	1	0	0	0	0	0

→

```
100 DATA 10011000
110 DATA 10011000
120 DATA 11111111
130 DATA 00111101
140 DATA 00111100
150 DATA 00100100
160 DATA 00100111
170 DATA 00100000
```

La línea 10 activa el (MODE) módulo de alta resolución de gráficos. La línea 20 empieza el bucle FOR NEXT requerido para leer (READ) (linea 30) los valores binarios. Para explicar la linea 40 yo brevemente introduciré la función del SPECTRAVIDEO en el manejo de los números binarios:

```
10 SCREEN 1
20 FOR T=1 TO 8
30 READ A$
40 S$=S$+CHR$(VAL("&B"+A$))
50 NEXT T
60 SPRITE$(1)=S$
70 PUT SPRITE 0,(128,96),8,1 :GOTO 70
100 DATA ........
(* Use above data lines #100 to 170)
```

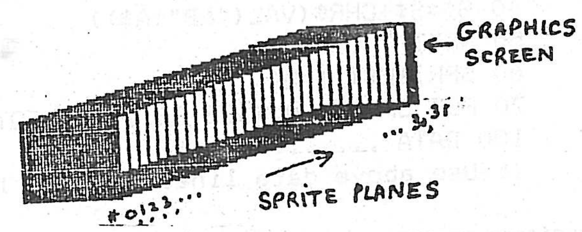
Las funciones incorporadas en el SPECTRAVIDEO poseen la habilidad de conversión entre los números decimales (regulares) y los valores binarios. Para poder decirle a su Ordenador que Vd. está especificando en número binario, deberá incorporar el prefijo &B antes del número. Entonces la expresión binaria de 11101 es expresada como &B11101 y el SV responderá imprimi·

endo 29., lo cual es el decimal equivalente del binario 11101. Para verificar esto Vd. puede usar la funcion BIN$,la cual es la contraria de &B por la que convierte un número decimal a un número binario. Trate PRINT BIN $(29). Mejor que escribir el prefijo &B en cada número binario en la linea de datos y READ (LEER) los valores dentro de una variable numérica. Loa autores del manual de usuarios han optado por leer (READ) cada número binario dentro de una variable "String" (A$) y entonces añadir el prefijo &B en frente de la cadena. Esto nos ahorra tiempo en introducir el programa además de ahorrarnos memoria, pero introduce la necesidad de usar la función VAL para extraer el valor numérico de la cadena final. Ahora volvamos a la programación de la línea 40.

La linea 40 toma la cadena A$, la cual contiene los 1's y los 0's, añade el prefijo "&B", toma los valores numéricos de esta cadena y los convierte en un caracter simple de tipo ASCII (con (CHR $), entonces añade al final de la cadena S$.(el cual está vacío inicialmente). Para el tiempo en que el bucle está terminado S$ contiene los 8 caracteres (bytes) necesarios para definir un sprite.

La funcion SPRITE $ en la línea 60, convierte a 'S$' en un sprite colocando los valores decimales de cada byte en la cadena, en la posición apropiada en la TABLA DE PATRONES DE SPRITE del Vídeo.

El formato de la función SPRITE$ es SPRITE $P,(x,y),C,S, donde:
P = sprite números planos (0,31)
x = coordenadas horizontales de la pantalla (0,255)
Y = coordenadas verticales (0,255)
c = color sprite (0,15)
S = sprite (0,255 o 0,63)
Para mejor entendimiento de los 'planos' de los sprites lo mejor es imaginarnos 32 hojas transparentes de plastico.



Cada una de estas está superpuesta una encima de la otra, y cada una puede tener un sprite dibujado en él. Un sprite en el plano más exterior (0) puede cubrir o superponer cualquier otro sprite en un plano más bajo (1-31), etc. Vd. puede usar estos factores para hacer que los sprites se muevan "en frente de " "detrás de" otros sprites, dependiendo en qué plano se encuentre el sprite

```
620 PRINT:PRINT" Quizas no haya raices": GOTO 52Ø
630 X2=C
640 RETURN
650 INPUT" Limites";X1,X2
660 GOTO 4ØØ
670 RESUME NEXT
```

Este programa es muy útil para el estudio de las funciones matemáticas y de su trazado por complicadas que sean.

En las funciones que posean asíntotas no se produce error de Overflow ni división by cero. El error es detectado en la linea 280 y se solventa con el RESUME de la línea 670 que hace continuar el bucle con el valor siguiente.

El espesor de los puntos que se pide en la línea 160 puede oscilar entre 0.1 y 5, aunque no necesariamente.

La escala de la representación puede oscilar entre 1 y 50, permitiendo que funciones muy "apretadas" puedan ser ampliadas.

Jugando con valores de espesor y escala podemos estudiar cualquier función de modo perfecto.

Si no quedamos satisfechos con la representación, hagamos un CTRL STOP y pulsando directamente la tecla de función 1, introduzcamos los nuevos parámetros.

Unos ejemplos ilustrativos son:

DEF FNY(X)=X*SIN(X), Intervalo -1Ø,1Ø; Espesor:1; Escala:1Ø.
DEF FNY(X)=X^3/(X^2-1), Intervalo -4,4; Espesor:Ø.1; Escala:15
DEF FNY(X)=X^2-3*-X-4, Intervalo -3,5; Espesor:Ø.5; Escala:1Ø

La búsqueda de raices en principio toma los límites usados para la función. En el caso de que la función tenga varias raices debemos suministrar los nuevos límites en los que creamos que hay otra raiz según la representación gráfica. Por ejemplo podemos buscar las raices de Y=XSIN(X) entre -7 y 7 viendo que hay varias raices.

Cuanto más exactas queramos que sean las raices tanto más pequeño debe de ser el error máximo que suministremos: Ø.1 será de una décima, Ø.Ø1, centésima, hasta lo que queramos: Ø.ØØØØ1 por ejemplo.

En el anterior Boletín nos referimos a un manual más detallado, traducido por INDESCOMP. Después de infructuosas gestiones no hemos logrado ningún tipo de condición especial para los socios del CLUB, así que el precio especial que anticipábamos no es tan especial.

Precio del Manual PVP .... 2.900,-Ptas.
Socios Club .... 2.500,-Ptas.

PROGRAMAS PROGRAMAS PROGRAMAS PROGRAMAS

PROGRAMAS PROGRAMAS PROGRAMAS PROGRAMAS PROGRAMAS

LISTADO DE REPRESENTACION DE FUNCIONES Y BUSQUEDA DE RAICES.

```
 10 REM Joaquín Ortíz, 1984, Representación de funciones.
 20 KEY2,"LIST 17Ø"+CHR$(13)
 30 KEY1,"RUN 15Ø"+CHR$(13)
 40 SCREENØ,Ø: COLOR 15,4,4
 50 PRINT "REPRESENTACION DE FUNCIONES"
 60 LOCATE1Ø,1Ø: PRINT"1- Representación."
 70 LOCATE1Ø,13: PRINT"2- Busqueda de raices."
 80 LOCATE22,22: INPUT" Pulse opcion";OP
 90 ON OP GOTO 1ØØ,32Ø
100 CLS
110 PRINT:PRINT:PRINT" PULSE LA TECLA DE FUNCION 2":?:?:?
120 ?:?:?" PONGA LA FUNCION":?:?
130 PRINT" PULSE TECLA DE FUNCION 1":?:?:?
140 STOP
150 CLS: INPUT" Extremos del intervalo";x1,x2
160 PRINT:INPUT"Espesor de los puntos";PE
170 DEF FNY(X)=
180 PRINT:PRINT:INPUT"Escala (1-5Ø)";ES
190 SCREEN 1
200 LINE(128,Ø)-(128,192),15
210 LINE(Ø,96)-(256,96),15
220 LOCATE125+ ES,92:PRINT"(left graph"A")"
230 LOCATE125-ES,92:PRINT"(left graph"A")"
240 LOCATE126+ES,1Ø2:PRINT"1"
250 LOCATE12Ø-ES,1Ø2:PRINT"-1"
260 FOR I=X1 TO X2 STEPØ.1*PE
270 PSET(128+ES*I,96-1.5*ES*FNY(I))
280 ON ERROR GOTO 67Ø
290 NEXT I
300 LOCATE2ØØ,17Ø:PRINT"Pulse una": LOCATE2ØØ,18Ø:PRINT"tecla"
310 IF INKEY$="" THEN31Ø ELSE 4Ø
320 REM Obtencion de raices por Teorema de Bolzano.
330 CLS
340 FOR I=1 TO 6:PRINT:NEXT I
350 PRINT " RAICES POR BOLZANO"
360 REM Las raices se encuentran en un intervalo
370 REM en el que haya cambio de signo
380 ?:?:?:INPUT" Error menor que";E
390 CLS
400 REM
410 Y1=FNY(X1):Y2=FNY(X2)
420 IF Y1=Ø THEN PRINT"Hay una raiz en "X1: GOTO 52Ø
430 IF Y2=Ø THEN PRINT"Hay una raiz en "X2: GOTO 52Ø
440 GOSUB 54Ø
450 XØ=(X1+X2)/2
460 YØ=FNY(XØ)
470 IF YØ=Ø THEN PRINT"Hay una raiz en"XØ:PRINT: GOTO 52Ø
480 IF ABS(X2-X1)<2*E then PRINT XØ " Es raiz con error menor que "E:
    GOTO 52Ø
490 IF SGN(FNY(X1))=SGN(YØ) THEN 51Ø
500 X2=XØ: GOTO 45Ø
510 X1=XØ: GOTO 45Ø
520 PRINT: INPUT"Otros limites (s/n)";Z$
530 IF Z$="s" THEN 65Ø ELSE END
540 REM Estudio del cambio de signo
550 IF SGN(Y1)<>SGN(Y2) THEN 64Ø
560 L=X2-X1
570 FOR I=X1 TO 1ØØØ
580 C=X1+L*RND(1)
590 IF SGN(Y1)<>SGN(FNY(C)) THEN 63Ø
600 NEXT I
610 PRINT" No se encuentra cambio de signo"
```

correspondiente. Las gráficas de pantalla más comunes son dibujadas debajo de todos estos planos de sprites, para que todos los sprites se puedan desplazar a los gráficos producidos por los comandos gráficos (CIRCLE, LINE, DRAW, ETC.). sin necesidad de alterarlos.

La coordenada horizontal 'x' puede ser cualquier valor entre 0 y 255. Usando valores mayores o menores dentro de este rango causará el que uno de los sprites se salga de la pantalla y aparezca en el lado opuesto de la pantalla.

La coordenada vertical "Y" también puede estar entre los valores 0 y 255; ahora, cualquier valor mayor que 191 pondrá el sprite en un lugar fuera del campo de la pantalla y por lo tanto no podrá ser visto (de todas formas el sprite está en existencia y activo) Existen dos posiciones "Y" debajo del 191, las cuales pueden ser de aplicación útil.

Usted puede "apagar" un sprite colocando en cualquier punto con una coordenada de un valor de 209. Cuando un sprite se encuentra situado en la línea del pixel 209 sin tomar en consideración la posición horizontal, el Ordenador considera ese sprite como inactivo o "apagado".

Otra coordena "Y" útil es la 208. Cuando Vd. coloca cualquier sprite en este punto sin tomar en consideración la posición horizontal, entonces todos los sprites serán apagados y borrados de la pantalla. Si Vd. mueve un sprite hacia abajo encima de esta linea Vd. notará que la pantalla parpadea secuencialmente en el momento en que el sprite pase sobre la línea pixel 208.

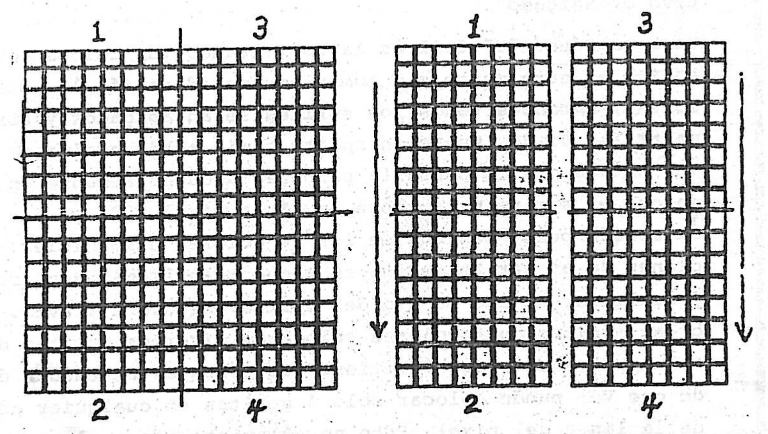
Un sprite puede ser uno de los 16 colores disponibles. Las porciones de un sprite que no se encuentran llenas son transparentes y cualquier objeto debajo del sprite, saldrá a través. Vd. puede crear sprites multicolores colocando diversos sprites de colores uno encima del otro. Vd. tiene que trabajar alrededor del factor de que Vd. puede colocar solo 4 sprites en cualquier coordenada "y" de la linea del pixel. Esto no sólamente afecta la superimposición de sprites pero también aquellas que están horizontalmente paralelas en la pantalla.

Por cada 8 x 8 pixel sprites, el número sprite ('s') puede ser cualquier valor entre 0 y 255. La memoria RAM del SV tiene la capacidad de acumular 256 patrones de sprites diferentes de 8x8 al mismo tiempo, sin embargo solamente 32 de éstos pueden ser proyectados en la pantalla de un golpe. Cuando se utilizan los pixel sprites de 16x16 solamente 64 patrones de sprites pueden ser acumulados en la memoria Video Ram. y 'S' es entonces cualquier valor entre 0 y 63. (yo digo) 'valor en vez de número' porque se puede utilizar cualquier decimal , octal, hexadecimal, o número

Cuando se están utilizando los pixel sprites de 8x8 se puede además especificar si se desea el tamaño del sprite que sea aumentado o regular, los cuales poseen la misma forma pero son de doble tamaño. Vd. puede lograr ésto añadiendo un segundo parámetro al comando de la pantalla. Para especificar sprites de alta resolución use SCREEN 1,0, (esto es equivalente a SCREEN1). Screen 1,1 especifica el modo de alta resolución con los sprites aumentados. Vd. puede especificar también los tamaños del sprite en el modo de baja resolución con el SCREEN 2,0 ó SCREEN 2.1 (Vd. notará que los sprites estan afectados por modo de SCREEN y no hay tal cosa como sprites de baja resolución.

Un sprite 8x8 aumentado es del mismo tamaño que un sprite de 16x16 en tamaño regular. Dasafortunadamente Vd. no puede mezclar los sprites aumentados y los no aumentados en la misma pantalla. Sus sprites tienen que ser,o todos aumentados, o todos no aumentados.

OK! Ahora vamos a observar a los sprites de 16x16. En vez de usar solamente una caja de bloques 8x8 Vd. usa 4 cajas. Otra vez Vd. rellenará los cuadrados de acuerdo a sus necesidades. Cuando ha llegado el momento de transpasar estos cuadrados a números binarios Vd. debe hacerlo en orden correcto.



Primeramente , Vd. dividirá la caja 16x16 en dos partes, verticalmente a través del centro. Empezando en la esquina izquierda (cuadrados 1&2), Vd. puede trabajar hacia abajo convirtiendo el patrón en 1's y 0's. Esto producirá el primer conjunto de 16 números binarios.

Entonces empieza en la parte superior de la esquina derecha, (cuadrados 3 y 4) y trabaje en descenso convirtiendo este patrón a números binarios. El resultado de esto será la conversión del segundo conjunto de 16 números binarios (Vd. debe poseer ahora 32 números binarios). Coloque estos números en el orden que se efectuó la conversión, en las líneas de datos. Claro que Vd. ne-

RUTINA DE CODIGO MAQUINA DE TRANSFERENCIA.

33, 126, 234, 6, 255, 14, 130, 237, 179, 33, 125, 235, 6, 255, 14, 130, 237, 179, 33, 124, 236, 6, 255, 14, 130, 237, 179, 201,0,0 .

MATRICES DE LOS CARACTERES:

```
0,0,0,0,0,0,0,0
32,32,32,32,0,0,32,0
80,80,80,0,0,0,0,0
80,80,248,80,248,80,80,0
32,120,160,112,40,240,32,0
192,200,16,32,64,152,24,0
64,160,64,168,144,152,96,0
16,32,64,0,0,0,0,0
16,32,64,64,64,32,16,0
64,32,16,16,16,32,64,0
32,168,112,32,112,168,32,0
0,32,32,248,32,32,0,0
0,0,0,0,0,32,32,64
0,0,0,248,0,0,0,0,
0,0,0,0,0,96,96,0
0,0,8,16,32,64,128,0
248,136,136,152,152,152,248,0
16,16,16,48,48,48,48,0
248,136,8,248,192,200,248,0
248,136,8,248,24,152,248,0
136,136,136,248,24,24,24,0
248,128,128,248,24,152,248,0
248,136,128,248,152,152,248,0
248,8,8,24,24,24,24,0
112,80,80,248,152,152,248,0
248,136,136,248,24,24,24,0
0,0,32,0,0,32,0,0
0,0,32,0,0,32,32,64
24,48,96,192,96,48,24,0
0,0,248,0,248,0,0,0
192,96,48,24,48,96,192,0
112,136,8,16,32,0,32,0
112,136,8,104,168,168,112,0
112,80,80,248,200,200,200,0
240,208,208,248,200,200,248,0
248,136,128,128,192,200,248,0
248,136,136,200,200,200,248,0
248,128,128,248,192,192,248,0
248,128,128,248,192,192,192,0
248,136,128,216,200,200,248,0
136,136,136,248,200,200,200,0
32,32,32,48,48,48,48,0
8,8,8,8,200,200,248,0
160,160,160,248,200,200,200,0
128,128,128,192,192,192,248,0
252,148,148,212,212,212,212,0
248,136,136,200,200,200,200,0
248,152,152,136,136,136,248,0
248,136,136,248,192,192,192,0
248,136,136,136,152,152,248,0
240,144,144,248,200,200,200,0
248,136,128,248,24,152,248,0
248,32,32,48,48,48,48,0
136,136,136,200,200,200,248,0
136,136,136,216,216,216,248,0
148,148,148,212,212,212,252,0
136,136,248,32,248,200,200,0
136,136,248,96,96,96,96,0
248,136,8,16,32,200,248,0
112,64,64,64,64,64,112,0
0,0,128,64,32,16,8,0
112,16,16,16,16,16,112,0
32,80,136,0,0,0,0,0
0,0,0,0,0,0,248,0
64,32,248,0,0,0,0,0
0,0,248,8,248,200,248,0
128,128,248,136,200,200,248,0
0,0,248,136,192,200,248,0
8,8,248,136,200,200,248,0
0,0,248,136,248,192,248,0
112,64,64,240,96,96,96,0
0,0,248,136,200,248,8,248
128,128,248,136,200,200,200,0
32,0,32,48,48,48,48,0
8,0,8,8,24,24,152,248
128,144,160,192,248,200,200,0
32,32,32,48,48,48,56,0
0,0,252,148,212,212,212,0
0,0,248,136,200,200,200,0
0,0,248,136,200,200,248,0
0,0,248,136,200,248,128,128
0,0,248,136,152,248,8,8
0,0,248,136,192,192,192,0
0,0,248,128,248,24,248,0
64,64,240,64,96,96,96,0
0,0,136,200,200,200,248,0
0,0,72,72,104,48,48,0
0,0,148,212,212,212,252,0
0,0,136,248,32,248,200,0
0,0,136,248,32,96,96,96
0,0,248,136,16,224,248,0
24,32,32,64,32,32,24,0
32,32,32,0,32,32,32,0
192,32,32,16,32,32,192,0
64,168,16,0,0,0,0,0
```

Se trata simplemente de una transferencia desde la RAM a la RAM de vídeo. Se transfieren sucesivamente los bloques que comienzan en EA7E, ED7B, EC7C.

La instrucción VPOKE(2815),VPEEK(2815) se utiliza como puntero de la RAM de vídeo, y por ello debe cursarse inmediatamente antes de la llamada al código máquina.

Desde luego los caracteres que he propuesto son sólo un ejemplo y pueden hacerse todos los cambios que uno pueda imaginar.

Me gustaría aclarar que en este programa los caracteres que se redefinen no son los caracteres normales sino los inversos. Por esta razón se puede cambiar de un juego a otro con el sólo cambio de video normal a inverso y viceversa. Dada la rapidez del cambio del juego de caracteres, gracias al código máquina, nada impide redefinir los normales e inversos al mismo tiempo e incluso tener en memoria en distintos bloques varios juegos completos que pueden implantarse cuando se desee, con sólo llamar a una rutina de máquina diferente.

Por último, hacer notar que el uso de SCREEN destruye los caracteres redefinidos y restaura los normales y los inversos convencionales. Claro está que los están en la RAM permanecen allí y pueden implantarse de nuevo en la RAM de vídeo para tener de nuevo los caracteres redefinidos.

Si lo que se desea es redefinir sólo algunos caracteres para usarlos como caracteres especiales (para juegos usando la pantalla cero, o como símbolos matemáticos, por ejemplo) es aconsejable que los caracteres a redefinir sean los gráficos (con códigos entre 160 y 211) para poder obtenerlos con la simple pulsación de una tecla. Para redefinir uno de estos caracteres se hará lo siguiente: se introduce en una sentencia DATA la matríz del carácter que deseamos introducir ; se escriben las siguientes líneas:

```
100 FOR I=1 to 8
110 READ A: VPOKE 1791 + I + 8*(ASC("carácter") + 64), A
120 NEXT
200 DATA Matriz del carácter a introducir.
```

Donde "carácter" es el carácter sobre el que vamos a poner el nuevo. Por ejemplo sería el carácter LEFT GRAPH de la tecla A si quisiéramos cambiar este carácter por el nuevo. Para terminar, un ejemplo: cambiar el carácter left graph de la tecla N por una "ñ". Poner como DATA:248,0,176,200,136,136,136,0

Cuando se haya cargado el programa desde cassette debe hacerse CLEAR 800,60000 para proteger la zona de almacenamiento del programa.

Como habíamos mencionado anteriormente , ahora Vd. se encuentra limitado a 64 patrones de sprites diferentes en su vídeo RAM. (numerados del 0 al 63). Por lo demás las instrucciones son las mismas que para los sprites 8x8, excepto para especificar cualquier magnitud de tamaño.

Para poder crear y usar los sprites 16x16 Vd. debe colocar un '2' ó '3' en el comando de la pantalla "SCREEN COMMAND", dependiendo de si desean sprites 16x16 aumentados o regulares. Screen 1,2 se refiere a alta resolución sin estar los sprites aumentados, y SCREEN 1,3, se refiere a sprites aumentados en el modo de alta resolución. Para los sprites 16x16 en modo de baja resolución use SCREEN 2,2 ó SCREEN 2,3,.

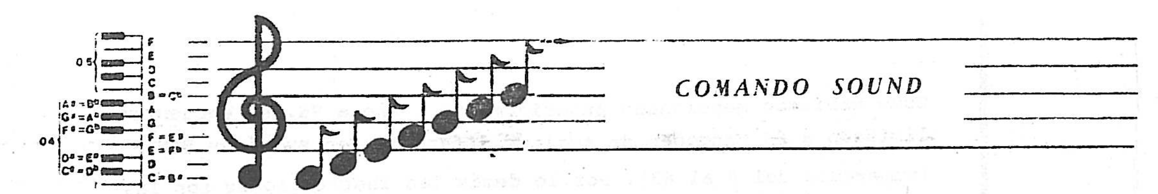
Más adelante explicaremos las funciones varias de los sprites, cómo los sprites pueden ser utilizados de manera diversa, etc. Trataremos de clarificar en los próximos artículos.

ASUNTO WIDTH

Cuestión referente a si es posible obtener un WIDTH de menos columnas creo que se verá satisfecha si prueba a poner el número de columnas que se desee en la dirección 62787. Por ejemplo:

POKE 62787, 20 Funciona como el width pero en este caso sí se obtienen las 20 columnas. Ni qué decir tiene que pueden usarse cualquier número de columnas desde 40 hasta cero.

Si se desea utilizar sólo media pantalla puede ponerse en la otra mitad alguna tabla o texto que se desee utilizar mientras escribimos cualquier programa. Por ejemplo, si hacemos POKE 62787,15 reservamos 25 columnas a la derecha que pueden servir para mantener permanentemente cualquier información en esta zona. Debe tenerse la precaución de poner esta información antes de cursar el comando poke que fija el número de columnas. Además no debe hacerse CLS ó SCREEN si no queremos ver desaparecer lo que habíamos puesto allí. No hay manera de escribir nada en esa zona y está a salvo de listados (el listado se hace más estrecho). La única manera de entrar en esa zona es el VPOKE. Para volver al modo normal debe hacerse WIDTH(39).



Posee la forma SOUND n° de registro, número

El ordenador posee 16 registros de los cuales sólo 14 (0-13) son utilizables.

N° registro

0 ....Canal 1. Ajuste fino de frecuencia. Rango 0-125
1 ....Canal 1. Ajuste ordinario de frecuencia. Rango 0-15
2 ....Canal 2. Ajuste fino de frecuencia. Rango 0-255
3 ....Canal 2. Ajuste ordinario de frecuencia. Rango 0-15
4 ....Canal 3. Ajuste fino de frecuencia. Rango 0-255
5 ....Canal 3. Ajuste ordinario de frecuencia. Rango 0-15
6 ....Ajuste de la frecuencia predominante. Rango 0-31
7 ....Mezclador de canales
8 ....Canal 1. Volumen Rango 0-15 }
9 ....Canal 2. Volumen " 0-15 } Poner 16 para lograr envolturas
10.......Canal 3. Volumen " 0-15 }
11.......Ajuste fino de la envoltura. Rango 0-255
12.......Ajuste ordinario de envoltura. Rango 0-255
13.......Tipo de envoltura.
14.......I/o del A }
15.......I/o del B } ¡No utilizar¡

Los seis primeros registros nos permiten obtener un ajuste de cada canal mediante dos registros por canal.
Nos permiten obtener pues 16x255 = 4080 tonos diferentes.
Con el registro n° 6 seleccionamos una frecuencia predominante en el sonido.
Los registros 8 a 10 poseen dòble acción controlando el volumen de cada canal o en el caso de que su rango sea 16 nos permite lograr una envoltura en el canal seleccionado.
Los registros 11 y 12 se utilizan para el ajuste de la envoltura que se selecciona mediante el registro 13.
Los tipos de envoltura son:

0,1,2,3 ó 9
4,5,6 ó 7
8
10
11
12
13
14
15

teres se deben cursar de nuevo las instrucciones:

VPOK 2815,VPEEK(2815):DEFUSRØ=60000:?USR(Ø)

Si se desea conservar permanentemente el nuevo juego de caracteres es preciso interceptar el STOP y los errores con una línea como la siguiente:

```
Ø  STOP ON: ON STOP GOSUB 1ØØØØ: ON ERROR goto 1ØØØØ
....
1ØØØØ  SCREEN Ø:
1ØØØ1  VPOKE 2815,VPEEK(2815):DEFUSR Ø =60000: ? USR(Ø): ON
       ERROR GOTO Ø
```

NOTA IMPORTANTE:

Es importante el hecho de que la instrucción de restaurar los caracteres tanto la primera como las demás veces, debe cursarse con una sola línea, es decir: VPOKE 2815,VPEEK(2815):DEFUSRØ=60000:PRINTUSR(Ø) No debe hacerse por separado porque entonces no saldrá lo que queremos.

Los propietarios de un SVI 318 deberán sustituir la dirección 60000 por otra de su elección, siempre teniendo en cuenta que no está permitido invadir la zona reservada al sistema (los tres últimos Kbytes). Además será preciso sustituir cuatro de los códigos de la parte de máquina. Para hacer este cambio hágase lo siguiente: una vez elegida la zona donde se va a guardar el conjunto de caracteres, se toma la primera dirección y se obtiene su expresión hexadecimal con PRINT HEX$ (dirección). Saldrán cuatro dígitos.Tómense los dos primeros dígitos por la izquierda y póngase en decimal con PRINT &Hnm donde nm son los dos dígitos tomados. Saldrá la expresión decimal correspondiente a esos dos dígitos. Tomar este número y ponerlo en el tercer lugar de la parte de código máquina, exactamente donde pone 234. Tomar ahora los dos dígitos restantes de la expresión hexadecimal, pasarlo a decimal y ponerlo en la segunda posición, donde pone 126.(NOTA: A la dirección elegida hay que sumarle 30 antes de todo este proceso). Ahora sumamos 795 a la dirección elegida y repetimos el proceso sustituyendo esta vez en los lugares 12ª y 11ª (donde pone 235 y 125 respectivamente). Por último sumamos 54Ø a la dirección elegida y repitiendo el proceso sustituímos en los lugares 21ª y 2Øª (donde pone236 y 124 respectivamente.

Para los que entienden código máquina se trata de sustituir los valores de las direcciones de partida que se cargan en HL.

Los que tengan un SV 328 no necesitan hacer ningún cambio.

El listado de la parte de código máquina es el siguiente:

```
LD HL,EA 7E
LD B,FF
LD C,82
OTIR
LD HL,ED 78
LD B;FF
LD C,82
OTIR
LD HL,EC 7C
LD B,FF
LD C,82
OTIR
RET
```

que no se utiliza este programa como ayuda y he propuesto un pequeño programa de carga de las matrices diseñadas que se indican en la tabla adjunta. En esta tabla está contenido un juego completo de caracteres, sin incluir los caracteres gráficos que seguirán siendo los mismos.

Si tenemos paciencia y tiempo la entrada de todos estos números en el Ordenador, se realizará de la siguiente forma

- Meteremos todos los números, empezando por el código máquina en líneas DATA. La parte del código máquina la pondremos en una primera línea DATA separada del resto, para mayor claridad. En las restantes líneas pondremos 7 matrices completas, es decir, 56 números. Así vamos escribiendo líneas DATA hasta tener todos los números metidos en estas líneas. Catorce líneas DATA deberían ser suficientes.
- A continuación escribimos las siguientes líneas:

```
100 FOR I= 60000 TO 60135
110 READ A: POKE I,A: NEXT
```

Si todo está bien al hacer RUN obtendremos el mensaje de "Out of DATA"; si no es así es que hemos puesto datos en exceso.

Para hacer una última prueba hacemos PRINT PEEK(60000) que debe dar 33.Una vez hecho todo esto cogemos cualquier cinta virgen que tengamos por ahí y la ponemos en el cassette. Seguidamente escribimos el siguiente comando:

```
BSAVE "CARACT", 60000, 60800
(Para cargar hacer BLOAD"CARACT")
```

Lo que hacemos con esto es guardar el bloque de RAM en el que están los caracteres redefinidos en el cassette, para utilizarlos cuando queramos.

A continuación viene la prueba definitiva. Cursamos las siguientes instrucciones: (debe cursarse todo junto, sepsrado por el signo ":")

```
VPOKE2815, VPEEK(2815)
DEF USA Ø.= 60000: PRINT USA(Ø)
```

Si todo está bien lo primero que veremos será que las barras de las funciones han cambiado y ahora tenemos letras diferentes, éstas son las letras que habíamos diseñado. Esto está muy bien pero, ¿ Cómo utilizar ese nuevo juego de caracteres ?. Sencillamente, haciendo POKE 65077,1. Con este comando hemos cambiado el juego de caracteres normal por el nuevo , y cualquier cosa que escribamos utilizará los nuevos caracteres. Para volver a los caracteres normales hacemos POKE 65077,Ø.

Hay un pequeño problema y es que la redefinición de caracteres se pierde si hacemos un SCREEN. Para volver a restaurar los carac-

El registro n° 7 nos permite activar o desactivar los distintos canales de la forma siguiente:

128	64	32	16	8	4	2	1 ← Valor decimal de las posiciones
B7	B6	B5	B4	B3	B2	B1	B0
NO USAR		ACTIVA RUIDO			ACTIVA TONO		
SIEMPRE UNO		C	B	A	C	B	A

Se pone un 1 para los canales no activados y un 0 si el canal está activado.

Ejemplos:

SOUND 7, &B11100111 , activa el ruido en los canales A y B

SOUND 7, &B11110110 , activa tono y ruido en el canal A

```
10 SOUND 7,&B11111100
20 SOUND 8,15:SOUND9,15
30 FORN=1 TO 255
40 SOUND 0,N:SOUND 2,256-N:FORI=1 TO 20:NEXT
50 NEXT
60 GOTO10
```

```
10 SOUND 7,&B11111100
20 SOUND 8,15:SOUND9,15
30 FORN=1 TO 255
40 SOUND 0,N:SOUND 2,256-N:FORI=1 TO 20:NEXT
45 FORJ=1 TO 20
47 SOUND 3,J:FORI=1 TO 20 :NEXT
50 NEXTJ,N
60 GOTO10
```

```
10 REM---Demostracion de los tipos de
20 REM---envolturas
30 CLS
40 SOUND 7,&B11111110
50 SOUND8,16:SOUND0,150:SOUND12,10
60 FORI=1 TO 15
70 SOUND13,I
80 SOUND12,10
90 PRINT"ENVOLTURA NUMERO"I
100 FORN=1 TO 1500:NEXT
110 NEXTI
120 SOUND8,0
```

COMANDO PLAY

NOTAS MUSICALES	SIGNOS	
C - DO		
D - RE	+	semitono más alto
E - MI	#	semitono más alto
F - FA		
G - SOL	-	semitono más bajo
A - LA		
B - SI	.	tiempo y medio

Letra O . Rango 0 a 7 es decir ocho octavas

Letra V . Control de volumen. Rango 0-15.

Letra L . Control de la longitud de las notas. Rango 1 a 255. Para notas largas utilizar rango bajo y viceversa.

Letra R . Control de pausas o silencios

Notas	Tiempo	Longitud	Silencios	Tiempo	Longitud
	1/8	64		1/4	32
	1/4	32		1/2	16
	1/2	16		1	8
	1	8		2	4
	2	4		4	2
0	4	2			

Letra T . Ritmo total de la música. Rango 32 a 255. El 32 es el más rápido y 255 es el más lento

T120 es el tiempo usual y ordinario

Letra N . Rango 0-84. Cada número corresponde a una nota de cada octava

Letra S y M. S controla la forma del sonido y va casi siempre acompañado por M que nos distingue los diferentes tonos. El rango de S es de 0-15 y M toma valores de 1 a 65535 Aunque existen 16 valores para S sólo existen 9 ondas distintas descritas en las envolturas.

Cabe resaltar que si se está trabajando con una forma y tono determinados seguirán activando a lo largo del programa a no ser que sean modificados.

Para volver a la normalidad se teclea PLAY"M65535T255L63"

REDEFINICION DEL JUEGO DE CARACTERES

(basado en una idea de Javier Gonzalo)

El Spectravídeo permite la redefinición completa del juego de caracteres, ya que almacena las matrices en la RAM de vídeo y permite cualquier modificación en esta zona. En modo Screen Ø las matrices de caracteres se almacenan a partir de la dirección 1792 de la ram de vídeo. Para modificar un carácter basta con cambiar el contenido de los bytes que forman la matriz del carácter. Por ejemplo, si hacemos VPOKE 1792 +32*8,255 aparecerán líneas horizontales en toda la pantalla. Esto se debe a que hemos hecho que el primer byte que compone la matriz pase de ser cero a ser 255. (que sea 255 significa que todos los bits del byte están a uno, es decir están encendidos). Es fácil deducir que en condiciones normales el carácter "espacio" está compuesto por 8 bytes iguales a cero. Para volver a tener el espacio blanco hacer vpoke 1792 + 32*8,Ø. El número 32 de esta expresión no es más que el código ASCII del carácter "espacio". Así pues, si quisiéramos cambiar la letra "n" para que pase a ser "ñ" haríamos:

VPOKE 1792 + ASC("n")*8,248

Se multiplica por 8 porque cada carácter está definido por 8 bytes.

Antes de seguir no estará más decir que no puede seguirse esta norma para cambiar todos los caracteres porque para cambiar los caracteres gráficos (de códigos 160 a 215) debemos sumar 64 al código ASCII para hacer cualquier modificación. Esto se debe a que en medio están los caracteres inversos.

Otra consideración previa es que sólo los 5 primeros bits de cada byte se utilizan ya que el sexto es un espacio, necesario para separar las letras entre sí.(En realidad son los 5 últimos bits ya que el primero sería el de más a la derecha). Esto es así porque si no no cabrían 40 caracteres en la pantalla ya que su anchura máxima es de 256 puntos (240 en Screen Ø). No obstante pueden definirse 6 bits, pero teniendo en cuenta que los caracteres aparecerán pegados unos a otros.

Además el último byte de la matriz es cero para evitar que los caracteres se toquen por arriba y por abajo. Así pues, cada matriz de un carácter se reduce a 7 bytes de los que sólo se aprovechan 5 bits. Este hecho de que sólo se definan 5 bits hace que no haya ningún número mayor de 248 en las matrices del juego de caracteres. (Esto no será así en el caso de los caracteres gráficos, que en muchos casos aprovechan 6 bits).

Hechas estas aclaraciones resulta fácil deducir que para redefinir el juego de caracteres bastará con modificar mediante vpoke todas las matrices, haciendo un diseño previo de los caracteres que queremos conseguir. La mejor forma de hacer el diseño y almacenar las matrices una vez modificadas es utilizar un generador de sprites tal como el Sprite Generator. Este programa permite diseñar cómodamente los caracteres y almacenarlos sin ningún problema. En el ejemplo que propongo he supuesto